GENI Research Plan

GDD-06-28

GENI: Global Environment for Network Innovations

Version 4.5 of April 23, 2007

Status: Interim (Version 4.5)

This is the last version to be released under auspices of the GENI Research Coordination Working Group and the GENI Planning Group. The GENI Science Council will handle future releases of this document.

This document was prepared by the Research Coordination Working Group, and the GENI Planning Group.

Authors:

David Clark, *MIT*

Scott Shenker, *UC Berkeley and ICSI*

Aaron Falk, ed., *USC ISI*

Contributors:

Lorenzo Alvisi, *Univ. Texas*

John Byers, *Boston Univ.*

Kevin Fall, *Intel Corp.*

Paul Francis. *Cornel Univ.*

Ramesh Govindan, *USC*

Joe Hellerstein, *UC Berkeley*

Kevin Jeffay, *Univ. North Carolina*

Jim Kurose, *Univ. Massachusetts*

John Mitchell, *Stanford Univ.*

Fred Schneider, *Cornell Univ.*

Salil Vadhan, *Harvard Univ.*

John Wroclawski, *USC ISI*

Ellen Zegura, *Georgia Tech*

This work is supported in part by NSF grants CNS-0540815 and CNS-0631422.

Revision History:

Version	Change log	Date
V 3.0	Original version posted	9/30/06
V 4 (4.3)	Reordered sections 3 and 4. Added section 2.2 (grand challenges). Major revision and expansion to section 3 (experiments). Complete revision of section 5 (requirements).	1/2/07
V 4.4	Inserted 9 sidebars Inserted preface Modified wireless and optical sections Added material to section 5 on getting users and the role of wireless kits Inserted appendix on non-research issues Front-to-back minor editing pass, including "signposting".	3/6/07
V 4.5	Minor modifications, and noted the forthcoming change in document ownership.	4/23/07

Table of Contents

About This Document

The case for GENI is, at its core, extremely simple and is based on the following four observations:

- **There are serious problems facing the Internet**: the Internet and the systems it supports face serious problems, such as inadequate security, reliability, manageability, and evolvability. There are also future opportunities the Internet may not realize because of its technical shortcomings. These problems and potentials are of great importance to society.
- **There are possible solutions to these problems**: the networking research community has proposals that address many of these concerns, and is actively working to develop additional approaches.
- **There are severe experimental barriers**: While our current tools – analysis, simulation, and small-scale experimentation – add to our understanding, they are not sufficient to fully evaluate the viability of new designs, and it is currently impossible to experimentally validate such designs under realistic conditions. The inability to determine which, if any, of the proposed approaches would work in practice is severely hindering scientific progress in the field, and makes deployment extremely unlikely.
- **GENI would transform the situation**: GENI would provide a facility where such experiments could be carried out, enabling proposals to be realistically evaluated. In so doing, GENI would transform the way science is done in this field.

This case, though conceptually simple, requires substantial elaboration, and that is the purpose of this document.

This document presents the case for GENI by addressing both *outcomes* and *approaches*. Outcomes are the long-term benefits that advances in networking and distributed systems could bring. These outcomes can be expressed as technical characteristics, such as security, reliability, and the like, or they can be expressed in terms of "grand challenges" that improved networks and distributed systems might play a crucial role in achieving.

Approaches are design alternatives being considered by the research community that might bring about these outcomes. The research community is largely unable to adequately test such design proposals now, and GENI is intended to fill this gap. To illustrate the nature of questions that GENI might address, this document describes a range of possible experiments. Many of these examples have been drawn from the first round of successful FIND proposals. The inclusion of these or any other particular approaches is not meant as an endorsement; they are merely illustrative of the variety of experiments GENI will support.

The case for GENI requires both desirable outcomes and feasible approaches, and the document splits its emphasis between these two perspectives. Because flexibility is one of its primary goals, GENI will enable experimentation on an extremely diverse set of approaches. These approaches could, in aggregate, enable a spectrum of outcomes far too broad to cover in a single document.

However, the essential case for GENI rests on areas where GENI will transform research, not merely augment it, i.e., on areas where GENI is *necessary*, not just *useful*. To that end, this document focuses on outcomes that revolve around a reconceptualization of the Internet. This

is an area of research that has great potential impact but is currently hampered by significant experimental hurdles. As such, this research thrust provides the strongest rationale for building GENI.

This narrowed focus of this document should not be seen as diminishing GENI's applicability to other important areas, notably distributed systems and applications. In fact, GENI could support experiments in a far broader set of disciplines, including the social sciences. These are important endeavors for which GENI could serve as an extremely useful experimental platform. However, because GENI is not seen as absolutely necessary for progress in those fields, this document does not dwell on those possible applications.

This document will be read by a wide variety of audiences, including the lay community, members of other scientific disciplines, researchers in computer science, and our colleagues in networking and distributed systems. To cover this spectrum, this document is intended to be both understandable to lay people and meaningful to experts in the field. Achieving the former has necessitated a long and fairly general description of the basic research questions in networking, as well as a discussion of the scientific nature of systems engineering; most of this material will be redundant to experts. Similarly, in order to bring some of the technical questions into sharper focus for the more expert audience, there are several "cut-outs" where specific technical approaches are described in more detail; unfortunately, these may not be accessible to all readers.

Lastly, this document is called the Research Plan. The MREFC process requires a Science Plan that addresses not only research, but also issues of education, outreach, international relations, and industrial participation. These non-research topics are briefly discussed in an Appendix, but a more in-depth treatment will be developed after the GENI Science Council has taken ownership of this document.

Executive Summary

Over the last two decades, the Internet has transformed the practice of science, the shape of business, and the life of people around the world. The speed of this transformation is breathtaking, as new applications emerge in months rather than years, and this transformation is not finished. In 10 or 15 years, our communications infrastructure and the services it supports will likely be materially different from today. The goal of this project—called GENI, for Global Environment for Network Innovations— and the research that it supports is to make sure that this next stage of transformation is guided by the best possible science, engineering and systems design, and that NSF-sponsored research plays a leadership role in the future.

This report argues for both a new research paradigm for networking and the allied research areas, and an experimental facility to support that research. As part of this argument, we summarize the requirements and limitations that will drive our global network and networked systems toward a different future, we describe the approach to research being put forward by NSF and the research community, we discuss the role an experimental facility will play in this research, and we catalog a set of prospective experiments and trials that will be performed using this facility.

The drive toward a future Internet

The Internet has been so successful that it is easy to imagine a rosy future just by extrapolating the present. However, there are aspects of its design, based on decisions made in the 1970's that severely limit its security, availability, flexibility, and manageability. These design limitations cannot be removed by minor incremental adjustment of the existing network, and if left unaddressed, will greatly hinder society's ability to utilize and exploit the Internet in the future.

For many years the network community's approach has been to address these limitations with a series of short-term "patches." Unfortunately, these patches have led to growing complexity, resulting in a system that is both less robust and increasingly difficult and expensive to operate. There is now a growing consensus in the networking research community that we have reached the stage where patching is no longer sufficient, and a fundamental rethinking of the Internet is required. In response to this assessment by the research community, NSF has evaluated its approach to funding research in this area, and concluded that a transformation of its research agenda is necessary.

GENI will support research that can lead to a revolutionary future Internet with greatly improved properties: better security, enhanced generality, better integration of wireless and advanced optical technology, integration with the future world of sensors and embedded processors, better techniques for network management and better options for the economic health of the industry sector. The research can lead to a new generation of sophisticated, highly distributed applications and application support services, dealing with issues such as location aware services, identity services, and new approaches to resilient and available services.

The NSF strategy for transforming research

The nature of Internet research is that innovation occurs at all scales. Some of the invention is "innovation in the small"—individual projects that help to scope out the landscape of the possible — but not all progress is of this sort. The key developments, such as the Internet itself and the World Wide Web, were coordinated, multi-year projects driven by a collective process of envisioning a future and then working to create it. Similarly, we believe that an Internet

fundamentally better than today's is not going to emerge by the incremental aggregation of many un-coordinated ideas but instead will require a more organized, coordinated attack on the problem, driven by an overarching view of what the future outcome should be.

The classic pattern of the NSF's Directorate for Computer & Information Science & Engineering (CISE) funding – single PI and small group grants – is well suited to "innovation in the small", but it is not, by itself, well suited to long-term, coordinated assaults on long-term objectives: "innovation in the large". To fill this gap, NSF is planning to augment its traditional funding model with new funding focus areas, new coordination mechanisms to move research from the small to the large, and a state-of-the-art experimental platform on which new ideas can be tested, deployed and evaluated.

The first focus research area solicited by CISE is FIND, or Future Internet Design, part of the recently completed NeTS solicitation[1]. This solicitation called for research in networking that was explicitly motivated by a vision of a future network, as opposed to a motivation based on incremental improvement of the present. It described a new model for doing collaborative research, which is expected, over the next three years, to lead to a small number of coherent proposals for a new future Internet, which can then be developed and tested on the experimental facility. A number of research projects from the FIND portfolio are described in this report.

The proposed experimental facility will allow researchers to experiment with alternative network architectures, services, and applications at scale and under real-world conditions. Through the use of virtualization, GENI will support multiple independent experiments running simultaneously across a diverse set of network technologies. GENI will also permit continuously running experiments, thereby allowing mature prototypes to support a live user community, which is essential for evaluating innovations under realistic conditions and for creating a population of users whose demonstrated interest in a new capability can stimulate technology transfer to the commercial sector. Through its extensive tools for measurement and data collection, GENI will both facilitate experimental research and provide a rich source of data to the larger research community. In sum, GENI will support a seamless research process for taking large-scale ideas—innovation in the large—from conception, through validation, to deployment.

The nature of experimental computer science

Why is a facility such as GENI needed as part of this new experimental paradigm? A future Internet will not be defined by one or two new features, but by the integration of a number of mechanisms into a cohesive whole, sometimes called an *architecture.* A candidate future Internet will then have to be evaluated relative to a large number of requirements, of the sort summarized above and detailed in section 2. Evaluating a balance among a set of multi-dimensional requirements is much harder and less precise than a simple optimization of a variable. The complexity of the evaluation is such that real-world experience with a running system is necessary, in order to detect unanticipated interactions of mechanism and requirements, validate expectations of utility, and explore the consequences of real users with unexpected objectives. Without exposing a system to real testing in a context as close as possible

[1] See http://www.nsf.gov/pubs/2006/nsf06516/nsf06516.htm

to the real world, the limitations of simple models and analysis may not be understood in a timely manner. The process usually proceeds iteratively, with first designs being subjected to evaluation that leads to revised and refined designs.

This need for experimental evaluation, of course, is not limited to networks in particular. Few ideas make their way to market without being prototyped and tested, whether the innovation is a new drug, a new car, or a new Internet. And it is not realistic to expect others to perform this job for the research community—the community has to prove the worth of its own ideas to get others to pay attention, and experimental deployment is a necessity to validate these ideas. GENI, by filling the gap between paper studies and simulation on the one hand, and broader impact on the other, empowers and motivates the research community to take up innovation in the large. Without this sort of facility, the broader vision of transforming the institution of research will be greatly weakened.

Another critical advantage of GENI as an experimental tool is that it can be instrumented for rich data capture and collection. The operational Internet today is not well instrumented, and what data is gathered is often not available to the research community because it is private and proprietary to the commercial service providers. Powerful measurement capabilities can both enhance the capabilities for experimental evaluation, and provide a rich source of information for the broader research community, including those who build models and perform more theoretical analysis.

What we expect to evaluate and demonstrate

The range of experiments anticipated for GENI are based on ongoing research, and as well prior research results going back at least 10 years, which have been proposed but not tested because there was no platform on which to try them out. The scope of research covers experiments on architectural building blocks—specific ideas that might be a part of a future architecture, and as well tests of complete proposals for a future Internet. Experiments also include new applications and application support services, and "grand challenges" for the research community.

The end-point of the anticipated research is some number of integrated proposals for a future global network. The research community has proposed a number of approaches to designing a future global network, each of which is a candidate to be evaluated using GENI. Here is a summary of several of them, to illustrate the range of thinking. Section 3 contains an elaboration of these ideas, and a richer catalog of proposed research topics.

- **A highly general and flexible global network based on virtualized resources**. Today's Internet assumes a single packet format, a single approach to routing, and so on. The virtualization alternative proposes that all we need to assume in common is that there are physical resources (links connected by processing elements) that can be virtualized, or sliced into shares that can be used by different sets of users for different purposes. In this view, there could, for example, be one packet format and routing scheme for information dissemination, another for real time communication, and perhaps a scheme for bulk data transfer that does not even employ packets.
- **A global network for information dissemination**. Today's Internet assumes the dominant communication paradigm is an end-to-end interactive exchange of packets in a point-to-point conversation between two machines. But most patterns of communication at the application layer do not follow this pattern. Email is forwarded in

a series of steps from server to server, web content is often downloaded from caches and relay points, and so on. The patterns of dissemination are often one-to-many, not one-to-one. So perhaps a future network should concentrate on a coherent architecture at this level, and allow a range of transport mechanisms to support it. In this scheme, as in virtualization, we need not agree on a common packet format, or even on packets, but in contrast to virtualization, the point of common agreement is "higher" than in the current Internet. These two ideas are complementary, not contradictory.

- **An architecture for global sensing**. If we accept that in 10 years, most of the computers will be small embedded processors rather than large, powerful processors, then a future Internet should be designed to support the application patterns of these devices. Perhaps the most challenging and important paradigm to support is global sensing, which involves integration and manipulation of data across the world, not in a locale.
- **An architecture for communication that is not real time.** Both of the previous ideas involve communication patterns that are not interactive end-to-end, but which proceed by stages, where information is positioned for rapid delivery, integrated, and then forwarded. One view is that this general paradigm, sometimes called Delay Tolerant Networking, may come to dominate the future Internet. (Even for telephony, it is often "not interactive", a phenomenon called "phone tag".)
- **An architecture that supports real-time communication with tight time bounds.** In contrast to the idea above is the proposal that a future Internet should support the option of bounded-delay real time interaction for such purposes as remote control, telephony and real time streaming, and so on.

The design of GENI is general enough that this full range of concepts can be developed, evaluated and deployed, in order to gain real world experience with the concepts. To realize these high-level architectural proposals, there are also a number of more specific ideas that need to be elaborated and proved. Section 3 provides an expanded description of these concepts, and as well a catalog of some of these more specific ideas, to convey the richness of the anticipated GENI research agenda.

A Case Study of Experimental Systems Research

Here is a story that illustrates the power of experimentation, innovation and discovery when the community is able to carry out large-scale experiments with real users. It describes work that was carried out on PlanetLab, a system that does not support the range of experiments that GENI will. But this project, like many others, has posed further hypotheses that will be evaluated using GENI

A researcher designed a new system for Content Distribution that he believed scales better under load, yet has response time that's comparable to the best-known techniques. Using the best methodology of the day, he simulated the system and quantified the potential improvement in aggregate throughput. He published a paper that reported 60-91% improvement over the state-of-the-art.

Then PlanetLab became available, which allowed the researcher to deploy the system in a realistic setting, at scale, with real user traffic. The researcher took advantage of the facility, and within days of deploying the system (v1), learned an important lesson: unanticipated traffic compromised the security of the system, making it unusable. The researcher deployed a redesigned system (v2) that took this lesson into account.

Within weeks of deploying the new system, the researcher discovered that performance was compromised by failures of the Domain Name System. Based on additional observations, the researchers discovered the root cause, and in response, demonstrated how the Content Distribution Network (which was designed to make web content more available) could be adapted to also make DNS resolution more robust. The researcher modified his system (v3) and deployed it on PlanetLab.

Based on instrumentation of this system, the researcher discovered that the best known models of DNS behavior were all wrong, and produced a new model that can used by other researchers.

Based on other data collected by instrumenting the system, the researcher discovered that he was able to observe two orders of magnitude more Internet failures than any existing observation platform has yielded. This resulted in a more accurate model of Internet behavior that other researchers are able to incorporate into their research.

The researcher also recognized that he could augment his original system (v4) to also diagnose Internet failures in real-time, and use this system to build adaptive applications that are able to route around failures, resulting in an even more robust service.

After gaining further experience, the researcher discovered that his system performs poorly when distributing big files, especially to a large set of clients, but that by including new mechanisms, he was able to redesign the system (v5) to yield large-object throughput that scaled with the number of clients that request the object. One of the more interesting lessons of this exercise is that the algorithms proposed by others to solve this problem do not work well in practice, and it is only by a thorough evaluation of various engineering tradeoffs that he was able to design a system (v6) with robust performance under a wide-range of conditions.

Epilogue 1: Researcher never bothered to return to the issue of the specific algorithms used in his original system, as they were in the noise relative to the other factors that actually influence an Internet service.

Epilogue 2: Researcher understood factors that influence network robustness at a deep level, and set out to create a clean-slate network architecture that incorporates these lessons in the core of the design. The new architecture is dissemination-oriented rather than client/server oriented, and thus must include completely new approaches to security because content is now decoupled from specific points (servers) in the network.

GENI will be the test platform to evaluate this new architecture.

1 The need for transformative research—the NSF initiative

The Internet has emerged as a consequence of early Federal research funding during the 1970s and 1980s (from DARPA and then NSF), which in turn inspired major commercial investment and technology deployment, finally leading to the penetration of the Internet into almost every aspect of society, government and the economy. The academic community has continued its tradition of research in networking and contribution to the future of the Internet. NSF is now the major source of academic funding for this sort of research in the United States, and this fact has motivated an assessment of how NSF funding drives the process of research and innovation, and how NSF can support the research community as a valuable player in defining the future.

The impact of research is often described as "innovation"—the transformation of research results into practical consequences. The nature of Internet innovation is that it occurs at all scales. Some of the invention is "innovation in the small"—individual projects that explore possible innovations that help to scope out the landscape of the possible. This sort of work is marked by the chaotic experimentation that results from independent innovation and a tremendous diversity of opinion as to where we are going. But not all progress is of this sort. The key developments, such as the Internet itself or the World Wide Web, were "innovations in the large": they resulted from coordinated, multi-year projects driven by a collective process of envisioning a future and then working to create it.

Looking forward, there are many aspects of the future that will benefit from a coherent vision of what this future might be. For example, we believe that a more secure Internet is not going to emerge by the incremental aggregation of many un-coordinated ideas. This has been the approach for the last 20 years. What is needed is a more organized, coordinated attack on the problem, driven by an overarching view of what the future outcome should be.

Designing a system such as a new Internet is not just the discovery of one or two breakthrough ideas. Even the singular idea of packet switching is only part of what defines the Internet. Creation of an architecture for a system is different from scientific discovery, and fits within the general domain of *systems engineering*. Systems engineering is a process that includes specification of requirements, invention of new approaches, and a complex process of trade-off and balance among different functional objectives, and as well among different stakeholders and constituents. Validation involves a process that first tests specific proposals for new mechanisms to better understand their limitations and relative advantages, and second tests more complete architectures to determine fitness of purpose within the multi-dimensional space of requirements.

Designing a new Internet is perhaps like designing a new airplane. Many innovations may be proposed, and the process of design must validate these innovations. But the success of the overall design is the integration of concepts to produce a design that balances a number of disparate considerations that include fuel efficiency, noise abatement, minimum required runway length, capacity, air safety and cost. In aircraft design, individual innovations become useful when a new plane is designed. The airframe market sees a steady introduction of new planes, which provides a platform for new ideas to enter the market. Similarly, there are some innovations in networking that can only enter the market if we contemplate a new Internet.

The classic pattern of NSF funding (single PI and small group grants) is well suited to "innovation in the small", but it is not, by itself, well suited to long-term, coordinated assaults on long-term objectives—"innovation in the large". The National Science Foundation directorate for Computer & Information Science & Engineering, together with the research community it supports, has concluded that the research community must be supported to do this larger sort of work: the community should work to have a few defensible visions of what our networked world should look like in 10 or 15 years, the community should identify those requirements that call for coordinated research and integration, and NSF should create a context in which this work can be carried out. To this end, NSF is planning to augment its traditional funding model with new funding focus areas for research, new coordination mechanisms to move research from the small to the large, and GENI, a platform on which new ideas can be tested, deployed and evaluated. The combination of funding, coordination and GENI positions the community to carry out what might, in other fields, be called "big science". We call it "innovation in the large".

As we will elaborate, GENI will support experimentation across a wide range of computer science and communications, including a demonstration of possible future Internets, new and innovative distributed applications and services, new tools to support these applications, and demonstrations of new communications technology.

1.1 Outline of this document

In section 2, we consider the question of whether it is important to take up the challenge of proposing a new global network for a time frame 10 or 15 years out. We catalog a set of objectives for a such a network, and argue that the case is compelling. This section is about *outcomes*, both technical and social.

In section 3, we switch to approaches and describe the range of anticipated research that can lead us to a future network. We catalog a set of experiments that might be done using GENI, a set of mechanisms and protocols, as well as complete systems, that can be the target of research once there is an experimental platform in place.

In section 4, we consider the nature of experimental research, and discuss why a platform such as GENI is required in order to carry out this sort of research. We outline a set of experimental methods that GENI can support, and argue that without a platform that can support a trial implementation, this line of research is impractical.

In section 5, we summarize a list of requirements that these various experiments imply for the design of GENI. We list the high-level requirements, describe a reference implementation for GENI derived from these requirements, and justify the specific design decisions embodied in the reference implementation.

2 Is it time to rethink the Internet?

It is a remarkable story. In a little more than twenty-five years, the Internet has gone from an obscure research network known only to the academic community, to a critical piece of the national communication infrastructure. To appreciate the significance of this transformation, consider that in 1989, a bug in the Internet's core routing algorithm inconvenienced a few thousand researchers. In 2003, the SQL slammer attack grounded commercial airline flights, brought down thousands of ATM machines, and in the end, caused an estimated one billion dollars in damage. As our dependency on the Internet grows, so do both the risks and the opportunities. This makes it imperative that we evolve the Internet to address new threats, accommodate emerging applications and technologies, and foster the spread of the network throughout the physical world. Thus, it is our goal to define a new generation of the Internet, a *Future Internet*, able to meet the demands of the 21st Century. Achieving this goal is of critical national importance.

The Internet has been so successful that it is easy to imagine a rosy future just by extrapolating the present. Since everything about computers just gets cheaper, won't the Internet just get so inexpensive that everyone can afford it? Will it not become so easy to use that everyone can master it? Will it not continue to deliver new value—new applications and services—so that everyone will want to connect? For a lot of reasons, the answer to these questions is: No!

Today's Internet, based on design decisions made in the 1970's, is very successful, and yet assumptions built into its design limit its potential. These design assumptions cannot be removed by minor incremental adjustment of the existing network, and if left unchecked, they will limit society's ability to utilize and exploit this new technology.

What are these limits?

- *The Internet is not secure.* We hear daily about worms, viruses, and denial of service attacks, and we have reason to worry about massive collapse, due either to natural errors or malicious attacks. Problems with "phishing" have prevented institutions such as banks from using email to communicate with their customers. Trust in the Internet is eroding.
- *The current Internet cannot deliver to society the potential of emerging technologies such as wireless communications.* Even as all of our computers become connected to the Internet, we see the next wave of computing devices (sensors and controllers) rejecting the Internet in favor of isolated "sensor networks".
- *The Internet does not provide adequate levels of availability.* The design should be able to deliver a more available service than the telephone system. Our future network should be robust and reliable enough to meet the needs of society in times of crisis.
- *The design of the current Internet actually creates barriers to economic investment and enhancement by the private sector.* For example, barriers to cooperation among Internet Service Providers have limited the creation and delivery of new services, including transport-level services such as enhanced quality of service (QoS), and applications such as Internet-based telephone service. A large number of specific problems with the Internet today have their roots in an economic disincentive, rather than a technical shortcoming.

- *The Internet was not designed to make it easy to set up, to identify failures and problems, or to manage.* This limitation applies both to large network operators and the consumer at home. Difficulties with installation and debugging of Internet in the home have turned many users away, limiting the future penetration of the Internet into society.

These limitations are deeply rooted in the design of the Internet. It is easy to overlook them because of the astonishing success of the Internet to this point. In the mere decade since the Internet left the research arena and entered the commercial world, it has substantially changed the way we work, play, and learn. There are few aspects of our life that aren't touched in some way by the Internet, and few (if any) technological developments have had such broad impact in such short time. *However, we may be at an inflection point in the social utility of the Internet, with eroding trust, reduced innovation, and slowing rates of uptake.*

For many years the network community's approach has been to work around these problems with a series of short-term "patches." Unfortunately, these patches have led to growing complexity, resulting in a system that is both less robust and increasingly difficult and expensive to configure, control, and maintain. There is now a growing consensus in the networking research community that we have reached the stage where patching is no longer sufficient, and a fundamental rethinking of the Internet is required [AND05].

As much as correcting these limitations is an imperative for action, it is equally important that the Future Internet foster rather than inhibit emerging applications and technologies. A future Internet that only does better what it already does today is a very narrow view of the future. Yet for a variety of reasons we detail below, the Internet today is poorly positioned to accommodate the likely applications of the future. To realize its potential, a Future Internet must enable and foster:

- A world where mobility and universal connectivity is the norm, in which any piece of information is available anytime, anywhere.
- A world where more and more of the world's information is available online—a world that meets commercial concerns, provides utility to users, and makes new activities possible. A world where we can all search, store, retrieve, explore, enlighten and entertain ourselves.
- A world that is made smarter—safer, more efficient, healthier, more satisfactory—by the effective use of sensors and controllers.
- A world where we have a balanced realization of important social concerns such as privacy, accountability, freedom of action and a predictable shared civil space.
- A world where "computing" and "networking" is no longer something we "do", but a natural part of our everyday world. We no longer use the Internet to go to cyber-space. It has come to us: a world where these tools are so integrated into our world that they become invisible.

We do not believe that a straightforward extrapolation of the current Internet will successfully reach this future world. The world as defined by computing and communications will be materially different in 10 years. The Internet will either deteriorate into a system where lack of trust has forced users into "online gated communities", and the Internet serves narrow needs such as e-commerce, or it will flower into a very different world, still open but more trustworthy, still accommodating to new uses, still growing and evolving, with opportunities

for continued innovation and the creation of new value. We conclude that now it the time to intervene and pick our future. Getting from where we are now to a new concept for an Internet is a goal of critical national importance. That is the motivation for this effort.

The research challenge at the center of this document is to understand how to design an Internet that achieves its potential. We characterize the research agenda along several axes. The first primarily focuses on issues of objectives—the requirements that will define the Future Internet. Section 2.1 presents the research agenda from the perspective of research challenges. The second axis, in Section 2.2, describes a number of "grand challenge" experiments, which help to signal how this proposed research could be of value to society. The third axis considers crosscutting foundational questions—modeling, analyzing, and formalizing the limits and properties of the Future Internet. Section 2.3 presents the research agenda from a foundational perspective. Note that we use these axes primarily for purposes of presentation; individual researchers typically pursue questions looking at these axes simultaneously.

2.1 Design Challenges and Opportunities

This section summarizes the important requirements and opportunities for the design of a Future Internet; these are the technical outcomes we hope to achieve. Determining how best to achieve these requirements and exploit these opportunities is the goal of the research to be enabled by GENI.

2.1.1 Security and Robustness

Perhaps the most compelling reason to redesign the Internet is to get a network with greatly improved security and robustness. The Internet of today has no overarching approach to dealing with security—it has lots of mechanisms but no "security architecture"—no set of rules for how these mechanisms should be combined to achieve overall good security. Security on the net today more resembles a growing mass of band-aids than a plan.

We take a broad definition of security and robustness. A traditional focus of the security research community has been on protection from unwanted disclosure and corruption of data. We propose to extend this to availability and resilience to attack and failure. Any Future Internet should attain the highest possible level of availability, so that it can be used for "mission-critical" activities, and it can serve the nation in times of crisis. We should do at least as well as the telephone system, and in fact better.

Many of the actual security problems that plague users today are not in the Internet itself, but in the personal computers that attach to the Internet. We cannot say we are going to address security and not deal with issues in the end-nodes as well as the network. This is a serious challenge, but it offers an opportunity for CISE to reach beyond the traditional network research community and engage groups that look at operating systems and distributed systems design.

Our most vexing security problems today are not just failures of technology, but result from the interaction between human behavior and technology. For example, if we demanded better identification of all Internet users, it might make tracking attacks and abuse easier, but loss of anonymity and constant surveillance might have a very chilling effect on many of the ways the Internet is used today. A serious redesign of Internet security must involve tech-savvy social scientists and humanists from the beginning, to understand the larger consequences of specific design decisions. This is one of several opportunities for CISE to involve other parts of NSF in this project.

We identify the following specific design challenges in building a secure and robust network:

- Any set of "well-behaved" hosts should be able to communicate among themselves as they desire, with high reliability and predictability, and malicious or corrupted nodes should not be able to disrupt this communication. Users should expect a level of availability that matches or exceeds the telephone system of today.
- Security and robustness should be extended across layers, because security and reliability to an end user depends on the robustness of both the network layer and the distributed applications.
- There should be a reasoned balance between identity for accountability and deterrence and privacy and freedom from unjustified observation and tracking.

2.1.2 Support for New Network Technology

The current Internet is designed to take advantage of a wide range of underlying network technologies. It is worth remembering that the Internet is older than both local area networks and fiber optics, and had to integrate both those technologies. It has done so with great success. However, there are many new challenges on the horizon.

Wireless: The current "new technology on the block" is wireless in all its forms, from WiFi today to Ultra-wideband and wireless sensor networks tomorrow. Wireless is perhaps one of the most transforming and empowering network technologies to come along, equal or greater in impact to the local-area network (LAN). For example, laptop sales exceeded those of desktop personal computers in 2003 and this trend towards compact and portable computing devices continues unabated. As of 2005, it is estimated that there are over 2 billion cell phones in use worldwide as compared with 500 million wired Internet terminals, and a significant fraction (~20%) of these phones now have data capabilities as 2.5G and 3G cellular services are deployed. In another 5 years, all cell phones will be full-fledged Internet devices implying inevitable changes both in applications and network infrastructure to support mobility, location-awareness and processing/bandwidth limitations associated with this class of end-user terminals. Clearly, we need to think now about how a Future Internet and new modes of wireless can best work with each other.

The most obvious consequence of wireless is mobility. We see mobility today at the "edge" of the network, when we read our email on our Blackberry or PDA. We have a weak form of mobile access with our laptops today, where we connect sporadically to WiFi hot spots. But the Internet itself does not support these activities well, and indeed in most cases is oblivious to them. The default node on the Internet today is still the stationary PC on a desktop. We must rethink what support is needed for the mobile host.

Perhaps less obvious, but equally important, while wire-based technology such as Ethernet just keeps getting faster, some wireless technology (especially that which works in challenging situations) is slow and erratic. The power of "always connected" may be accompanied by the limitation of unpredictable performance. We must think through how applications are designed to work in this context, and how a Future Internet can best support this wireless experience.

Similarly, because the devices connected to wireless networks must be power aware, and dynamic spectrum gives wireless devices an extra degree of freedom in how they utilize the communication medium, fundamental changes are needed in how we think about the network. The Future Internet must support adaptive and efficient resource usage, for example, by

treating links not as a rigid "input", but as a flexible "parameter" that can be tailored to meet the needs of the user.

Mobility increases the need to deal with issues of dynamic resource location and binding, and the linking of physical and cyber-location. In general, the network must support *location awareness*; the ability to exploit location information to provide services should be incorporated throughout the network architecture.

Finally, we need to understand the design principles for wireless networks in an Internet context. Like the Internet, the most popular wireless protocols today are insecure, fragile, hard to configure, and poorly adapted to support demanding applications. As just one example, the security of the popular 802.11 WiFi standard has been shown to be vulnerable to systematic attack [BOR01a]. We need to build realistic, live prototypes to point the way to addressing these fundamental problems with today's wireless technologies.

We identify the following specific design challenges in supporting wireless technology:

- A Future Internet must support node mobility as a first-level objective. Nodes must be able to change their attachment point to the Internet.
- A Future Internet must provide adequate means for an application to discover characteristics of varying wireless links and adapt to them.
- A Future Internet (or a service running on that Internet) must facilitate the process by which nodes that are in physical proximity discover each other.
- Wireless technologies must be developed to work well in an Internet context, with robust security, resource control, and interaction with the wired world.

Optical: A second technology revolution is taking place in the underlying optical transport, where the optics research community is about to undergo a dramatic shift, roughly equivalent to that of the electronics community in the early 1960s. Optical communications researchers are discovering how to use new technologies like optical switches and logic elements to deliver much higher performance at lower power than purely electronics solutions.

In particular, the advent of large-scale electronic integration that took the world by storm and led to the PC and wireless foreshadows a revolution that is about to take place with optics (photonics). The *photonic integrated circuit* (PIC) is allowing ever-increasing complexity in optical circuits and functions to be placed on a single chip alongside electronic circuits, to enable networking and communications paradigms not possible with electronics alone. As PIC technology matures, it will enable higher capacity networks that are reconfigurable, more flexible and have much higher capacity at much lower cost. This may involve moving from ring to mesh networks, from fixed wavelength allocations to tunable transmitters and receivers, from networks without optical buffering to ones with intelligent control planes and sufficient optical buffering, and from networks that treat fiber bandwidth as fixed circuits to networks that allow the fiber bandwidth to be dynamically accessed and utilized.

We identify the following specific design challenges in exploiting emerging optical capabilities:

- A Future Internet must be designed to enable users to leverage these new capabilities of the underlying optical transport, including better reliability through cross-layer diagnostics, better predictability at lower cost through cross-layer traffic engineering, and much higher performance to the desktop.

- A Future Internet must allow for dynamically reconfigurable optical nodes that enable the electronics layer to dynamically access the full fiber bandwidth.
- A Future Internet must include control and management software that allow a network of dynamically reconfigurable nodes to operate as a stable networking layer.

2.1.3 Support for New Computing Technology

The Internet "grew up" in the era of the personal computer, and has co-evolved to support that mode of computing. The PC is a mature technology today, and from that perspective, so is the Internet. But in 10 years, computing is going to look very different. Historically, when computing was expensive, many users shared one computer—a pattern of "many to one". As computing got cheaper, we got the personal computer—one computer per person. There was convenience and simplicity in the "one to one" ratio, and we have "stuck at one" for almost 20 years. But as computing continues to get cheaper, we are entering a new era, when we get "unstuck from one", and we have many computers to one person. We see the start of this transition, and the pace of change will be rapid. We can expect to be surrounded by many computing devices, supporting processing, human interfaces, storage, communications and so on. All these must be networked together, must be able to discover each other, and configure themselves into larger systems as appropriate.

In 10 years, most of the computers we deploy will not resemble PCs, they will be small sensors and actuators in buildings, cars, and the environment, to monitor health, traffic, weather, pollution, science experiments, surveillance, military undertakings, and so on. Today, prototypes of these computers are not hooked directly to the Internet but to dedicated "sensor nets", which are designed to meet the special needs of these small, specialized computers. A sensor net may in turn be hooked to the Internet for remote access, but the Internet is not addressing any of the special needs of these computers. It would seem odd if in 10 years we were still living with an Internet that did not take into account the needs of the majority of the computers then deployed. We should rethink now what we need to do to support the dominant computing paradigm 10 years from now. This will be of direct benefit to science, to the military, and to the citizen.

Sensor nets may seem very simple, and indeed because they are low-cost they avoid unjustified generality for application-specific features. But this technological simplicity and specificity does not mean that they do not have important architectural requirements. Sensors often have intermittent duty cycles, so they do not conform to the traditional end-to-end connectivity model of the classic Internet. Their design is driven by a structure that is data driven, rather than "connectivity driven". Some applications require a low and predictable latency to implement robust sense-evaluate-actuate cycles. A range of considerations such as these should be factored in to a Future Internet.

We identify the following specific design challenges:

- A Future Internet must take account of the specialized networks that will support future computing devices, which will imply such architectural requirements as intermittent connectivity, data-driven communication, support of location-aware applications, and application-tuned performance.
- It should be possible to extend a given sensor application across the core of the Internet, to bridge two parts of a sensor net that are part of a common sensing application but

partitioned at the level of the sensor net. In the limit, a future Internet should support sensing at a global level.

2.1.4 New Distributed Applications and Systems

The new networking and computing technologies described in the previous sections provide an unprecedented opportunity to deliver a new generation of distributed services to end-users. The convergence of communication and computation, and its extension to all corners of the planet down to the smallest embedded device, will enable us to provide users a set of services anytime anywhere, invisibly configured across the available hardware. The key enabling factor to these new services is programmability at every level—the ability for new software capabilities to self-configure themselves out over the network.

Today, we are seeing the first steps towards this future, where rich multimedia person-to-person communication is the norm rather than the exception; where every user becomes both a content publisher and a content consumer with information easily at our fingertips yet with digital rights protected; where the combined power of end host systems enables whole new paradigms of parallel computation and communication; and where the myriad of intelligent devices in our homes and offices become invisible agents on our behalf, rather than just another thing that breaks for no apparent reason and with no apparent fix.

Although the precise structure of these new applications and services may seem nebulous today, enabling their discovery is likely to be one of the most profound achievements of GENI. A common, reliable infrastructure can enable the research community to set its sights higher, rather than having to reinvent the wheel. Perhaps the best example of this is the history of networking research itself. When the first packet switched networks were developed, the intended target application was to support remote login by scientists to computing centers around the country. The Web wasn't on the radar, but the Web would have been much more difficult to invent without the Internet.

One design challenge is to understand how to build these new distributed services and applications. Engineering robust, secure, and flexible distributed systems is every bit as complex and difficult as engineering robust, secure, and flexible network protocols. Without a way to manage this complexity, both networks and distributed systems end up being fragile, insecure, and poorly suited to user needs. And like networks, models for managing this complexity can only be validated by building systems for real use on real hardware.

Another design challenge is how the Future Internet needs to adapt to support this new generation of distributed services and applications. The basic data carriage model of the current Internet is end-to-end two-party interaction. Early Internet applications grew up with just this form: two computers talking to each other—a remote login or a file transfer between two machines. But applications of today are not that simple. They are built using servers and services that are distributed around the network. The web takes advantage of proxies and mirrors, and email depends on POP and SMTP servers. There is a rich context for these servers—they are operated by different parties, often as part of a commercial relationship; they are positioned around the network in a way that exploits locality and variation in network performance; and they stand in different trust relationships with the end-users—some may be fully trusted and some (such as devices to carry out wiretap) have interests that are adverse to those of the users.

The original Internet design does not really acknowledge this complexity in application design. In fact, the Internet provides little support for application and service designers, and leaves to them much more of a design challenge than is appropriate. Today's more complex applications would benefit from a richer and more advanced set of application-support features. The Internet provides no information about location or performance—any application that needs this information must work it out for itself, which leads to lots of repetitive monitoring traffic (e.g. PING). The Internet reveals nothing about cost—if there is distance sensitive pricing, there is no online way for the application to determine this and optimize against it

Similarly, the current Internet is conceptualized at the level of packets and end-points. Both the low-level addresses and the Domain Name System identify physical machines. But most users do not think in terms of machines. They think in terms of higher-level entities, such as information objects and people. The Web is perhaps the best example of a system for creating, storing and retrieving information objects, and applications such as email or instant messaging capture both information and people in their design. But none of these applications require, as a fundamental requirement, that one user concern himself with what specific computer is hosting one of these higher-level entities. So there is a mismatch between the service of the Internet, defined in terms of physical devices, and the needs of the user, defined in more abstract terms of services.

As a part of a Future Internet, we should include architectural considerations at these higher levels: should people have identities that cross application boundaries? What are the right sorts of names for information objects? How can we find objects if the name does not specify the location? There are many such questions to be asked and answered. But perhaps the more basic question is: once we propose answers to questions at this higher level of conceptualization, is the service interface of the current Internet (end-to-end two-party interactions) the right foundation for these higher level concepts, or will a Future Internet have a different set of lower-level services once we recognize the real needs of the higher levels?

We identify the following specific design challenges:

- A Future Internet needs to develop and validate a new set of abstractions for managing the complexity of distributed services that can scale across the planet and down to the smallest device, in a robust, secure, and flexible fashion. This must include an architecture or framework that captures and expresses an "information-centric" view of what users do.
- A Future Internet must identify specific monitoring and control information that should be revealed to the application designer, and include the specification and interfaces to these features. For example, the Future Internet might reveal some suitable measure of expected throughput and latency between specified points.
- A Future Internet should include a coherent design for the various name-spaces in which people are named. This design should be derived from a socio-technical analysis of different design options and their implications. There must be a justification of what sort of identification is needed at different levels, from the packet to the application.

2.1.5 Network Management

The term "management" describes the tasks that network operators perform, including network configuration and upgrades, monitoring operational status, and fault diagnosis and repair. The

original design of the Internet did not fully take into account the need for management, and today this task is difficult and imperfect, and demands high levels of staffing, and high skill levels for those staff.

Network management is not just a problem for commercial Internet Service Providers. Any consumer who has tried to hook up a home network, only to have it fail to function, and has faced the frustration of not knowing what to do, has seen the limits of Internet management. Management at the user level is part of usability, and usability is a key to further penetration of the Internet into the user base. And corporations and institutions—any organization that runs Internet technology—suffer from the same management problems. The problem is endemic, and intellectually very hard to solve.

Better management tools are also vital to the goal of better availability. It has been estimated [YAN02] that perhaps 30% of network outages today are due to operator error. We cannot build a truly available network unless we deal with the problem of management.

A more sophisticated approach to management may depend on more powerful automated agents to support human decision-making. This is an opportunity for CISE to include researchers in artificial intelligence and machine learning as a part of this project.

We identify the following specific design challenges:

- An operator of a network region should be able to describe and configure his region using high-level declarations of policy, and automatic tools should configure the individual devices to conform.
- A user detecting a problem should have a tool that diagnoses the problem, gives feedback to the user in meaningful terms, and reports this error to the responsible party, across the network as necessary.
- All devices on a Future Internet should have a way to report failures.

2.1.6 Economic Well-being of the Internet

The Internet has evolved from its roots as a government-funded research project to a commercial offering from the private sector. Internet Service Providers, or ISPs, provide the basic packet carriage service on which all the other services and applications in the Internet depend. The centrality of the private sector links the future of the Internet to economic investment, which requires us to focus on an important truth: technical design choices can have a profound impact on industry structure. For example, the routing protocol that connects different ISPs together, BGP, allows certain patterns of interconnection and the expression of certain business policies. An early alternative was much more restrictive, and would have only worked if there was a single monopoly provider. The designers of BGP intentionally chose to avoid this outcome, a signal that they were becoming increasing sensitive to issues of industrial structure as it related to technical design. Any redesign of the Internet needs to consider how to encourage economic progress—the ongoing ability of industry to accommodate new advances while providing reliable service to customers.

Importantly, there are issues lurking in the current industry structure that presents barriers to progress. Two important barriers are the commoditization of the open IP interface and interconnection among ISPs. The open IP interface implies that anyone, not just the ISP, can offer services and applications over the Internet. This openness has been a great driver of

innovation, but the ISP may not necessarily benefit from this innovation. If all they do is carry packets, competition may drive the price of ISP service to the point where the ISP revenues do not justify upgrades and expansion. This tension can be seen today most clearly in the case of residential broadband. It also underlies the trends away from total openness to a world in which the ISPs block certain applications, and try to reserve to themselves the right to offer others. Problems of this sort have led to recent FCC intervention in the Internet.

Interconnection will always raise issues, because the ISPs that must interconnect may also be fierce competitors. In the traditional telephone carriers, problems of interconnection proved so difficult that regulators defined the rules. So far, this aspect of the Internet has avoided regulation, but the problems are real. Whenever a new service, such as end-to-end quality of service, requires ISPs to negotiate jointly about how to offer and price the service, that new service may not happen.

It is very hard for a set of companies positioned within an industrial structure to collectively shift that structure. But if we can conceive of a slightly different structure that removes some of the current impairments, this may be a powerful inducement to adapt our ideas to the betterment of both users and the industry serving those users. This is an area where NSF can encourage participation in our effort from other disciplines, such as economics and business.

We identify the following specific design challenges:

- Routing protocols must be redesigned to deal with the range of business policies that ISPs want to express. Issues to be considered include signaling the direction of value flow, provisioning and accounting for higher-level services, dynamic pricing, explicit distance-sensitive pricing, and alternatives to the simple interconnection models of peering and transit.
- A Future Internet must provide a means to link the long-term resource provisioning problems at one level to the short-term resource utilization decisions (e.g. routing) at higher levels.

2.1.7 The larger societal context of the Internet

The future of the Internet will perhaps be shaped most strongly by the considerations and concerns of the societies in which it is embedded. Technological innovation is useless and ineffective here unless it is accepted and exploited by users. For example, a fundamental question facing the design of a Future Internet is how to balance privacy against accountability. To what extent should users be anonymous as they use the network, versus what rights does society have in holding users responsible for their actions. To what extent will a future Internet empower its end users, the network operators, large corporate actors or the state? To what extent will control be centralized or decentralized? To what extent can a future Internet foster a free and open society? Should that be an explicit goal? The requirements in this space are uncertain, imprecise, and critical.

2.2 Grand challenges

The desiderata listed in Section 2.1 are largely technical in nature, and are rooted in our understanding of the Internet and its current limitations. However, this section approaches the issue quite differently, by focusing on several "grand challenge" objectives. These high-level objectives are offered as examples of the way a different network of the future might have a material impact on society. We argue that it is hard to imagine how to make any coherent

progress on these challenge questions without some sort of actual deployment and demonstration, as would be supported by GENI. In this way, GENI can not only help develop new architectural principles, it can also contribute to the accomplishment of "grand challenge" objectives.

2.2.1 Service in times of disaster

The Internet has grown up from its initial public sector funding to be a creature of the private sector, and this has happened at a time when in most countries the governments are deregulating their telecommunications operators. As a result, the services and functions the Internet offers are driven by private sector priorities. A great deal of attention has been paid to better security in support of e-commerce; but other applications with less economic payoff have received less attention. A very important example of a collective social need is service in times of crisis. The future Internet should helpfully and gracefully cope with disasters, both large and small**.** We are now all too familiar with the threat and impact of large-scale disasters on society. Whether these disasters are natural, as in hurricanes Katrina and Rita during the 2005 season, or human-caused, as in the terrorist attacks of September 11, 2001, they produce a crisis of enormous magnitude with impact on human lives and property. Large-scale disaster preparedness is incredibly important [NRC04].

The Internet has tremendous potential as a tool for citizen access to information, emergency notification and to provide access to emergency services. The telephone system provides E911, and newer services such as reverse 911. These were conceived and designed in an era when voice was the only mode of communication. What could the equivalent services be for a multi-media network like the Internet? Could a Future Internet tell citizens of a tsunami or a tornado, based on their location? Could a Future Internet provide reliable and trustworthy information during a terrorist attack? There is tremendous potential here, but it will not happen in any organized way unless it is designed and implemented. This sort of public-sector social requirement should be a first-order goal for a Future Internet.

A defining characteristic of many disasters is that there is a desperate and immediate need for information so that individuals can react as appropriately as possible (e.g., leave a location or stay put), so that first responders can communicate with victims and each other, and so that authorities can coordinate overall response efforts. Unfortunately, many disasters also damage or destroy large parts of the traditional communication infrastructure, so that exactly when there are pressing needs for reliable and trustworthy communication, the infrastructure is destroyed.

Even if the infrastructure survives, the network may not be adequate to meet the special needs of disaster response As an example, individuals desperate for information on the morning of September 11, 2001, inadvertently overwhelmed popular on-line news outlets such as CNN [CSTB03b, WIKI06] .

We identify the following specific design challenges:

- A Future Internet should be able to allocate its resources to critical tasks while it is under attack and some of its resources have failed. (For example, it should support some analog of priority telephone access that is provided today.)

- Users should be able to obtain information of known authority in a timely way during times of crisis. The network (and its associated applications) should limit opportunities for flooding, fraudulent and counterfeit mis-information, and denial of service.
- Users should be able to obtain critical information based on their location, and request assistance based on their location.
- Portions of the Internet should be usable by the attached computers even if they are disconnected from the rest of the network.

What approaches might contribute to building a network that can meet this goal?

The citizens must be able to receive trustworthy information, even in the context of a man-made disaster in which dis-information is a goal of the attacker. So attention to security, especially in the context of information security, is critical. Priority of access is critical, so past and future work on quality of service (QoS) must be incorporated.

Dealing with destruction of installed infrastructure is a problem that requires specific attention. Current research on rapid deployment of ad hoc wireless networks is clearly relevant, but there are many more problems. There has been less attention on the extended problems of keeping emergency infrastructure running and making it highly reliable. The research challenges involved in creating a functional, critically reliable communication system go beyond inventing a new routing protocol or adding radios to a network. These challenges span hardware, network architecture, and software; many would exist even if the physical infrastructure existed already. For example, hardware must be extremely portable, affordable, robust, and consume little power. Since communication may be intermittent and erratic, protocols that support disruption tolerant networking (DTNs) are critical to this objective. Specific research described in Section 3 support these objectives.

A primary challenge for these networks is operation without large teams of highly trained specialists, which implies low-cost automatic management, so the work described in this report on network management is an important building block, as is the work on wireless systems. Management must be cross-layer, from the physical layers up: the replacement infrastructure must be self-managing and self-organizing to a much larger degree than today's networks. Both initial setup and ongoing maintenance and upgrades must be simple, fast, and safe. Vastly reduced management complexity would ultimately reduce the impact of the most significant cause of unavailability---human error.

Management in times of disaster and disruption raises requirements that might not be as significant during normal operation.

- It should be possible for parts of the network to operate while partitioned from the rest of the network. This raises issues of address management, routing, and name resolution.
- It should be possible for a device (or a region of the network) that has crashed and lost all of its dynamic state to rejoin the network in a way that is both secure and as automatic as possible.

An ambitious challenge—post-disaster infrastructure: In the days following a significant disaster, a team of five people deploy and operate for at least six months a replacement network infrastructure that provides all necessary communication services to those within its coverage area. This deployment should take no more than three days to cover a small city of 20,000

people, providing equivalent services to the telephone network, cable network, and conventional Internet services. The deployment must be robust to challenging environmental conditions, unreliable power and transit infrastructures, and malicious activities such as the widespread looting that often occurs following significant disasters. It must support both commercial and life-critical emergency communication. The subsequent operation and upkeep of the network must require at most two people.

2.2.2 New visions of the personal cyber-experience

The futurist's story of personal technology has been told many times—an immersive experience where the individual is surrounded by computing and communications technology, both on the body and in the spaces through which he move. But what is all that technology for? Can we pose and test some larger stories about the benefit of this future to the individual, and the collective society in which the individual sits?

Ubiquitous health care: By using a combination of advanced networks in the home and advanced wireless network, we can imagine a world in which a person is always connected to a suitable health support system. Urban mesh networks can revolutionize health care within a metropolitan area by allowing medical practitioners and support staff to remotely monitor patients. Conversely, patients can have greater access to health-care educational resources. Such a system can greatly reduce the frequency of hospital visits, and deliver quantifiable economic benefits to the community. Married to a system for location-aware computing, this sort of system can provide greatly enhanced emergency access to medical services, with the potential of saving lives.

Participatory urban sensing: Mobile devices and platforms (e.g. cars) can be leveraged as a platform for data gathering in cities. This will drive development of network architecture through applications that have civic and cultural significance, such as participatory urban planning, community documentation, or tracking of environmental concerns across wide urban areas. Participatory sensing can track highly dynamic phenomenon, such as traffic congestion and road closures, or more serious crisis situations. If issues of privacy and misuse can be mastered, this sort of infrastructure can enhance the sort of notification and participation associated today with Amber Alerts and other mechanisms that reach out into the citizenry.

This vision has many implications for research. For example, a privacy-preserving data sharing specification framework is essential for participatory sensing. Beyond this, an architecture of mediating access points and routers that support *selective sharing* based on privacy policies is necessary for resolution control of location and time context, provisions for operating on physical context information based on the sensor readings (adding jitter/noise to data on-the-fly to decrease resolution), and so forth. Furthermore, naming and dissemination services that respect sharing policies, and services that manage coverage estimation and sampling of these autonomous mobile nodes are examples of novel components that participatory sensing requires.

Dealing with personal data: Individuals can currently generate large amounts of data, including digital photographs, music, and documents. In the future, sensors embedded in the environment or worn on the body will provide additional, automated streams of personal data. Even more data is created when content is shared with others and contextual information is used to correlate related data sources.

The optimistic vision of this future is that the ability to collect, store, and query vast amounts of personal data can provide the equivalent of "perfect memory", allowing a non-expert user to recall any image they have seen, any word they have read, written, heard, or spoken any time such information is needed. The pessimistic vision is that the user will drown in a mass of information that is impossible to search, recover or utilize in any useful way. The research community should set itself the goal of mitigating the pessimistic outcome, and exploring positive outcomes to see which are actually useful.

The power of on-line personal information comes fully to focus when we contemplate the interactions among individuals in cyber-space. Personal data can be part of the formation of virtual communities that allow people to interact with each other based on shared interests or experiences. For example, virtual communities may be formed based on geographical proximity (people who live in the same neighborhood or share the same commute), or a common activity (such as watching the same TV show or attending a sporting event). A virtual community can provide permanent archival storage of content created and contributed by its members.

Many research challenges in networking, storage, and distributed system design must be addressed to enable the collection and management of vast amounts of personal data. Data collected over a lifetime should be safely archived in a manner that provides reliability in the presence of system failure and local disasters. Data should never be lost due to disk failures, worm/virus attacks, or careless operation of the system. Archived personal data should be ubiquitously available to its owner at any time in any location.

Collection and maintenance of personal data requires strong privacy guarantees to guard against misuse of archived data, which in turn implies an architecture to deal with identity, as well as privacy. Yet, controlled sharing, such as the delivery of medical information to a family doctor, should be supported. Strong authentication may be required to ensure that data collected through automated or semi-automated channels are genuine.

Tele-presence: For many years, we have hoped that better tools for interaction and collaboration at a distance would reach the point where they can substantially reduce our need to commute from home to work, and to travel from city to city. With our increasing demand for team projects, for the linking of many specialists to achieve complex goals, and our higher aspirations for productivity, the demands for travel seem as great as they were in the past, or indeed getting worse. But a material improvement of tools in this area could reduce our consumption of natural resources, free up our travel and commuting time, and contribute materially to quality of life. These tools can help mitigate problems as diverse as caring for the elderly, reducing work-related travel, and reducing the social pressures that cause population movement from rural to urban areas.

Good tools in this area will require innovation and collaboration at many levels. The network must be able to deliver high-bandwidth data reliably, with controlled latency. Access to the network must be ubiquitous and affordable. The tools must be designed so that they are appealing substitutes for face-to-face interaction, which calls for research that involves sociologists, human factors and HCI research, and other cross-discipline teams.

2.2.3 Understanding and Affecting the Planet in Real-Time

While sensor networking has been an active area of research for almost a decade, most work has focused on the use of this technology in embedded applications. For example, most current

sensor network deployments support a single primary application (e.g., monitoring water delivery for plants on a farm) used by a single primary user or user group (e.g., the farmer). However, as we look to the future, the hope is to use the same basic sensor networking technology to understand the interdependent dynamic systems of our planet—its climate, politics, populations, ecology, economy, etc. Sensor and sensor-like instrumentation can provide real-time data, and large-scale computation can combine these distributed data streams to extract global meaning to, for example, assist in energy management. This visionary global-scale application can drive innovation in resource management, security of information, and network design.

Section 3.1.3 describes an experiment to demonstrate a future network designed for sensing at a global scale. With such a capability, we can begin to contemplate how to turn this to the benefit of society. We could imagine automated sensing becoming part of a larger process of observation and control. For example, we could reduce domestic power consumption to deal with high demand or emergencies in one or another part of the country. We could involve industry in a program of dynamically managed pollution control to deal with changed in atmospheric conditions. We could include various sorts of sensing in our plans to help citizens cope with disasters, as was discussed above.

To make this world a reality, we need to deal not just with technical issues, but larger social and economic issues—question of privacy, trustworthy data from sensors, incentives for various stakeholders to participate, and so on. But these larger questions become real and actionable only when we can see the possible shape of emerging technical capabilities.

GENI provides a highly desirable infrastructure for making progress towards addressing the above challenge. First, it provides an opportunity to deploy sensor networks at significant physical scale. Second, it allows multiple independent deployments of sensors to interact in a single experimental infrastructure. Third, it enables the deployment of new communication and in-network processing primitives that are unique to the sensor networking research area. Finally and perhaps most importantly, GENI provides an infrastructure that can be accessed by real application developers and users.

2.2.4 Vehicular Networks

In ten years, the automobile will be a powerful platform for computing and communications. However, the nature of this platform is not clear. It could be an open platform, open to multiple makes of cars and open to third-party application developers. On the other hand, vehicular networks could emerge as closed system, dedicated to specific cars or to specific purposes. Research and demonstration of vehicular networks could show the power of different sorts of architecture, and help shape the future.

Vehicular networks can improve navigation safety using wireless car to car and car to curb communications to rapidly propagate unsafe road conditions, accident reports to oncoming cars, or report unsafe drivers in the proximity and imminent intersection crashes [QUI02]. These networks can also consume location aware resource services [CHE04, LEE06, ZHO05], and can be used as an emergency communications network. However, vehicular networks have very different characteristics from many of the other applications considered above: large scale, temporary network disconnections, correlation between motion patterns and performance, and rapidly changing connectivity to the fixed network infrastructure.

An ambitious goal for vehicle networks is real-time accident avoidance. This scheme requires advanced instrumentation of the car, so that it can observe its environment, and highly reliable communication among cars so they can plan common actions and maintain safe trajectories. This objective would depend strongly on some of the research proposed in Section 3.1.6 on real-time, reliable communications. The demand for tight timing derives from the very close physical tolerances in which we operate cars today, but it is worth noting that this sort of inter-vehicle communication for safe operation and accident avoidance is utilized today in the air, where planes do not operate with such close physical proximity, and need for communication need not be sub-second.

2.2.5 Networks for the developing world

A significant opportunity for the networking community is the deployment of information and communication technologies (ICT) in the developing nations. This is enabled by the coming convergence of long-range high-speed wireless technologies, which permit the deployment of widespread ICT without the requirement of laying wire for the expensive "last mile", and the widespread availability of cheap, rugged end-user devices within the means of people in the developing nations and well-adjusted to variable environmental conditions. This development promises great and positive social impact while offering not simply the opportunity but the requirement to rethink the basics of network services and protocols[BRE05].

The availability of both inexpensive wireless technologies and devices offer the opportunity for the citizens of the developing nations to jump several generations of ICT, going directly from analog wired circuit-switched connectivity (or none at all) to digital wireless packet-switched networks. However, these networks will be different from their counterparts in the industrialized nations. In particular, the following properties will hold:

- Last-mile connectivity will not be based on wireline broadband technology, but will generally be provided by some form of wireless technology, typically mesh networking, and often with collaborative peer-to-peer networks.
- Connectivity in the developing nations may be episodic, lossy, and/or have high latency. Applications and services based on delay-tolerant, content-based, and multi-path routing and transport are likely to be requisite features of such networking. In other words, intelligence and storage within the core of these networks will likely be a must—quite unlike the original Internet.

As an example of a transformative vision of technology for the developing world, we look at the One Laptop Per Child (OLPC) project, championed by Nicholas Negroponte of MIT. This project envisions massive network deployments in challenging environmental conditions with primitive on-site support and management. In this respect, it represents a dramatic break from classic developed world networks, which rely on pristine environmental conditions in carefully engineered privately owned data centers and NOCs. In the developing world, in many cases, the endpoints are the infrastructure.

The sheer scale of the OLPC effort is daunting. The typical deployment the project envisions is 50-500 laptops in a village or school, which form an ad-hoc wireless mesh network. The laptops are then connected to a gateway, which is connected by a backhaul link to the national education network (or commercial ISPs) and thence to the Internet. The *trial* deployment in most nations is expected to be 1,000,000 laptops with an expected distribution of roughly 10,000 sites/nation. The total run rate of manufacture is expected to be on the order of 165,000,000 units/annum, giving a total worldwide deployment of 500,000,000-1,000,000,000. These devices

will, once deployed, form the single largest class of devices (with reasonable screen size) connected to the Internet.

The networking challenges faced by the OLPC project dwarf anything that has ever been attempted. We detail the challenges here:

- **The Village Gateways form the largest self-managing network ever conceived**. The OLPC project envisions a ratio of 100 laptops per gateway, with a total deployment ranging from 1000 gateways in a small nation to almost 1,000,000 in China.
- **The OLPC Content Distribution Network is the largest ever undertaken**. At its peak, the OLPC network envisions simultaneous content distribution (national textbooks, software, web content) to several hundred million devices through several million gateways.
- **The Village Mesh Networks will be the largest ever undertaken**. No one has formed an operational (as opposed to experimental) mobile ad-hoc networks of the scale conceived by OLPC (50-500 machines), particularly in the presence of anticipated significant environmental challenges. Management of such a network–particularly autonomic management from a village gateway–is a challenge that has never been undertaken.
- **The OLPC project envisions the largest shared store ever conceived**. The OLPC devices, for reasons of cost, power consumption, and durability, do not contain a hard disk: 0.5-4GB of flash is the on-device store. Because the laptops might be lost or damaged, they must be backed up into the network. The current proposal is that individual machines act as backup for each other, which will likely be supplemented by a large national backing store.
- **The OLPC Project will do all this over the most heterogeneous network ever built.**

The One Laptop per Child (OLPC) project is a strong case study for the experimental work that we expect will be performed on the GENI test bed. OLPC will exercise many of the open scientific areas that GENI experimenters will explore, and illustrates the social utility of the project in a dramatic fashion. If we examine the list of challenges discussed here, the alignment with the GENI research agenda is apparent. We have the following research challenges that derive immediately from the various networking projects in developing nations for the GENI research community:

- autonomic management of large-scale distributed infrastructures,
- autonomic management of large mobile ad-hoc networks,
- provisioning and update of large scale distributed storage systems,
- large scale content distribution systems, and
- sensing and control of routing and transport over large, heterogeneous networks with high latency and link loss.

Networking projects in the developing world challenge both our technical capabilities and our social conscience. Collectively, they form a clarion call to our profession, our nation, and our civilization.

2.3 Foundational Challenges and Opportunities

As the research community pursues the design of a Future Internet that delivers increasing value to society, we expect many opportunities to address foundational issues will arise— questions of fundamental limits, richer models about network behavior, and new theories about the nature of complex communication systems. This section describes some of the unique opportunities this effort creates.

2.3.1 Theoretical Underpinnings

Communications systems such as the Internet and the telephone system (which is morphing to the Internet) are perhaps the largest and most complex distributed systems we have built. The degrees of interconnection and interaction, the fine-grain timing of these interactions, the decentralized control, and the lack of trust among the parts raise fundamental questions about stability and predictability of behavior. There is beginning to emerge some relevant theories of highly distributed complex systems, some of which have roots in control theory and some of which draw on analogies with biological systems. We should take advantage of this work in this redesign, to improve our chances that we come as close as possible to the best levels of availability and resilience. There may be other important contributions from the theory community, for example, the use of game theory to explore issues of incentives in design of protocols for interconnection among competing Internet Service Providers. This is a chance for CISE to engage members of the theory community in this program.

2.3.2 Measurement, Analysis and Modeling

Mathematical models and analysis of measurement data have provided key insights into the fundamental limits of today's Internet. We believe they will continue to play a crucial role in the research on a Future Internet, and in fact, the design of new network architectures should be amenable to modeling and measurement in ways that today's Internet is not.

There are many examples that illustrate how measurements and analytical models have shed light on the limitations of today's architecture, including the following.

- Analysis of Internet traffic measurements has shown that IP traffic is self-similar. The burstiness of the traffic on multiple time scales makes traditional queuing models a poor predictor of network performance. Moreover, transport protocols such as TCP affect traffic in ways that further complicate analytical modeling. Although statistical analysis techniques have shed some light on the key properties of Internet traffic, analytical models of Internet performance remain elusive. Work on a Future Internet should consider whether protocols and mechanisms can be designed to be amenable to analytical modeling, making it easier to provide predictable performance and behavior to end users.
- Numerous measurement studies have unveiled key properties of Internet traffic, performance, and topologies. However, many of these studies rely on inference from edge measurements. With the increasing size and commercialization of the Internet, these studies have become ever more difficult to conduct, and the generality and accuracy of the results more suspect. A Future Internet should include support for measurement as a first-class mechanism because of the importance of measurement in understanding and operating the network.
- End users and network operators have great difficulty detecting, diagnosing, and fixing performance and reachability problems. The networking research community has created tools for anomaly detection and root-cause analysis, but these solutions are forced to work with extremely limited data collected from remote vantage points in competing domains. Today's protocols were not designed with diagnosis in mind. Future theoretical work can quantify the fundamental limits on diagnosing problems in today's network and identify key features for a future architecture to support diagnosis.

- The Internet's inter-domain routing system does not necessarily converge, depending on how the many domains select and configure their routing policies to achieve their business goals. Analytical models have demonstrated these problems and explored the fundamental trade-offs between business autonomy and global network convergence. These results suggest that we need a new routing system that strikes a better balance between the global properties of the system and the needs of users and operators for autonomy. A solution may require a move away from the existing inter-domain routing protocol, which has evolved via incremental steps into an extremely complex protocol in recent years.
- Measurement studies and analytical models have demonstrated significant benefits that competing domains could achieve by cooperating in computing paths for network traffic. However, today's routing protocols do not provide sufficient means for neighboring domains to negotiate over the exchange of traffic. New research in game theory and inter-domain negotiation offer promising solutions that are difficult to realize in today's architecture. Insights from these studies can drive the creation of new architectures for evaluation.
- Existing protocols and mechanisms were designed without the network operator's goals in mind, leaving the operator with (at best) indirect control over the traffic flowing through a domain. Recent theoretical work has shown that selecting the best configuration of the intra-domain routing protocols is a computationally intractable optimization problem, even for the simplest of network objectives. In addition, robustness is difficult to achieve because small changes in parameter settings can lead to large changes in the flow of traffic. Other mechanisms, such as queue-management schemes, do not lend themselves to analytical frameworks that guide operators in setting the tunable parameters. A Future Internet architecture could have manageability in mind from the beginning, by having protocols and mechanisms that either adapt on their own to network conditions or present tractable optimization problems to network operators.

Measurement and models have already provided significant insight into the behavior of today's protocols and mechanisms, and their fundamental limitations. The design of a Future Internet offers a rich landscape of research problems, as well as a unique opportunity to create new architectures with measurement and modeling in mind from the beginning. The research communities that are concerned with theory, analysis and modeling can benefit from the rich capture and logging of data from GENI experiments. The desire to make GENI data available to the broader research community raises obvious and important issues of privacy, the rights of experimenters, and so on. These issues are not unique to this context, and will have to be addressed as part of the oversight and administration of GENI. However, the theory community may have specific contributions to make here, with tools for privacy-enhancing and anonymizing algorithms.

2.4 Opportunities at Community Boundaries

Many of the opportunities for innovation and discovery will happen at the boundaries of traditionally separate research communities. A Future Internet will cut across the networking community (which traditionally considers issues inside the network), the distributed systems community (which traditionally innovates on the design of robust services and applications on

top of the network), the mobile and wireless community (which traditionally considers problems at the edge of the network), and the optical communications community (which traditionally develops device technology upon which networks are built). Important contributions can also be expected from other parts of CS, ranging from operating systems to theory. The theory community, in particular, has been very interested in finding fundamental bounds to networking capabilities (ranging from performance to security and robustness) and in finding new ways to design and validate practical network mechanisms.

Wireless is perhaps the most transforming of the current network technologies, with its promise of ubiquitous connectivity, the potential to provide connectivity without the high cost of fixed wireline infrastructure, and the capability to hook new classes of inexpensive computing devices such as sensors and actuators. But these capabilities challenge the Future Internet to deal with issues of mobility, new forms of routing (in which links are not pre-defined circuits but can be reconfigured in real time), and the problems of links with highly variable capacity.

Distributed systems and applications have traditionally been designed to run "on top of" the Internet, and to take the architecture of the Internet as given. This re-design raises the opportunity to better understand and assess higher-level system requirements, and use these as drivers of the lower layer architecture. In this process, mechanisms that are implemented today as part of applications may conceivably migrate into the network itself, and the relevant research communities themselves may blend together and share or exchange research ideas and architectural proposals.

Optical technology has proved itself as the workhorse of high-speed low-cost circuits that efficiently transmit data over long distances. However, there is the opportunity for optical technology to be used for more than simple, static point-to-point circuits; in the future circuits, ring and mesh networks will be configured dynamically using optical switch hardware managed by the same software as the electronic portion of the network. Even more exciting, there are new technologies just around the corner that will allow the optical fiber bandwidth to be dynamically accessed by edge nodes in a way that is as revolutionary to networking in the core as wireless has been at the edge. However, to realize this potential, the network architecture will have to be redesigned to take the emerging optical capabilities into account. Optical systems will be able to provide highly reconfigurable connections, which implies, for example, changes in the way a Future Internet will do routing. Promising directions in optical system design must be a driver for a Future Internet and mechanisms to integrate and manage this new technology in a new Internet architecture must be provided.

2.4.1 Broader Interdisciplinary Implications

Beyond looking across boundaries that separate technical sub-communities within Computer Science, this effort will benefit greatly from looking for help from disciplines much farther afield, disciplines as diverse as economics, sociology, and law. The larger societal context in which the Internet is embedded calls out for study from scholars from law, economics, and other of the social sciences and humanities. Collaboration with these sorts of researchers can help to broaden our understanding of the implications of our designs, and improve the chances that our work is relevant and successful.

3 An agenda for research using GENI

Section 2 described several outcomes derived from improved networking and distributed systems that would have a substantial and beneficial impact on society. The research community is busily developing technical approaches that would help realize these (and other) desirable outcomes. GENI will provide the substrate upon which these new architectural proposals, and new features and protocols, can be experimentally tested and evaluated.

This section describes a sampling of technical approaches currently under consideration. The inclusion of certain approaches, and the exclusion of others, should not be seen as implying any technical endorsement of one approach over another. The approaches described here, many of which were drawn from the first round of successful FIND proposals, are presented only as examples, and there are many other proposed approaches with equivalent merit.

This section first concentrates on overarching architectural proposals (in Section 3.1) and then delves into advances in the basic building blocks (Section 3.2), incorporating new network technology (Section 3.3), distributed systems (Section 3.4), and theory (Section 3.5), before tying these topics all back to architecture (Section 3.6). The intent here is to show that there is an active set of proposals already on the table that would require GENI for experimental evaluation. To cover the extremely broad spectrum of issues currently under investigation, this material is presented at a fairly general level. However, several "cut-outs" present specific approaches in more detail; these discussions may not be accessible to all readers.

3.1 Research on an Internet of tomorrow

While GENI can support a wide range of systems experiments, central to its justification is the conceptualization and demonstration of one or more proposals for an Internet for tomorrow. The payoff for all the research described here is the integration of new concepts into coherent overarching proposals for the future of networking and communications. Here is a summary of various research proposals that have already been brought forward as integrative visions, each of which might be demonstrated on GENI.

3.1.1 A global network with greatly enhanced generality and flexibility

Today's Internet assumes a single packet format, a single approach to routing, and so on. As an alternative, the **virtualization** concept proposes that all we need to assume in common is that there are physical resources (links connected by processing elements) that can be virtualized, or sliced into shares that can be used by different sets of users for different purposes. In this view, there could, for example, be one packet format and routing scheme for information dissemination, another for real time communication, and perhaps a scheme for bulk data transfer that does not even employ packets. As part of the FIND focus area, NSF has funded two proposals to explore this concept: *An Architecture for a Diversified Internet,* by Turner, Crowley, Gorinsky and Lockwood, and *CABO, Concurrent Architectures are Better than One,* by Feamster, Rexford, and Gao.

This concept raises many fundamental design problems and challenges. It creates a new layering, and results in two sets of industrial players: *infrastructure providers* and *service providers*. Network management must be rethought, since the responsibility for management must be divided among these players. Security must be rethought, since the infrastructure must be operated so as to meet the security needs of the most demanding service provider, without encumbering the service providers with weaker requirements. One of the most interesting

problems is how to ensure the economic health of this new industrial structure. A competitive market of interconnected infrastructure providers must emerge, unless infrastructure becomes a public sector responsibility. Planning for infrastructure investment becomes more difficult if the service planning is going on in a different firm. The right linkages between infrastructure and service providers must be created so that the infrastructure providers are motivated to install the facilities that the service providers actually need. Specific research questions include the division of responsibility for security and availability between the infrastructure and service layer, and the degree to which algorithms for virtualization may limit our ability to build service layers with tight real-time objectives.

The end point of this design will be the creation of two new interfaces, one that connects the infrastructure providers to each other, and one that connects the infrastructure providers to the service providers. The decisions taken about these interfaces will determine the ultimate success of the proposal, and will have to be tested in a running system if the idea is to receive sufficient validation.

One very exciting application of the virtualization concept is within a corporation or enterprise. Corporations, which today mostly operate their own intranets based on Internet technology, are turning to virtualization in their computer centers to allow for the rapid deployment of new services, migration between versions of services, and so on. One natural outcome of this trend might be the virtualization of all their communications facilities, so that they can deploy new distributed systems, perhaps based on different communications architectures, without the need for physical deployment of new equipment. The virtualization concept is particularly appealing in the enterprise context because in that case, the investment decisions for the infrastructure and service layers are being made by the same entity, so the planning process and economic justification is internalized within the firm.

3.1.2 A framework for managing information

Today's Internet assumes that the dominant communication paradigm is an end-to-end interactive exchange of packets in a point-to-point conversation between two machines. But most patterns of communication at the application layer do not follow this pattern. Traffic often does not flow directly from source to destination: email is forwarded in a series of steps from server to server, web content is often downloaded from caches and relay points, and so on. More importantly, the one-to-one communication pattern has been supplanted, current Internet traffic is overwhelmingly one-to-many or many-to-many. Peer-to-peer filesharing services distribute popular files among thousands of users simultaneously. (According to CacheLogic's study in 2004, BitTorrent accounts for one third of today's Internet traffic.[PAR04]) Streaming services distribute feeds of both live and on-demand media. Networked games need to quickly distribute state information among dozens or hundreds of players.

So perhaps a future design should concentrate on a coherent architecture at the level of information management and dissemination, and allow a range of transport mechanisms to support it. In this scheme, as in virtualization, we need not agree on a common packet format, or even on packets, but in contrast to virtualization, the point of common agreement is "higher" than in the current Internet.

Here are some of the specific problems that must be addressed to design this architecture.

Content Distribution At Scale: Schemes for content distribution have placed severe demands on the core Internet technologies, demands that many argue cannot be met using its current

design. Some research efforts have attempted to address these challenges by placing new functionality within the network to exploit localized knowledge and optimization opportunities. Some others have turned to overlay networks as a means to overcome the deployability challenges of the current Internet. Rather than attempting to change the underlying protocols, overlays aim to provide radical new services by layering them on top of the existing infrastructure. Irrespective of the approach, there has been only limited success and a core set of issues remain to be addressed.

- Efficient large-scale distribution: An ideal multi-point distribution mechanism should minimize bandwidth costs, avoid hotspots, utilize network resources efficiently, and exhibit quick response times, even in the face of unanticipated user demands. Though a few research prototypes and some commercial solutions have exhibited some of these properties under specific workloads, a general solution has not yet emerged and further research is required. Scaling effects have to be addressed along multiple fronts as both the amount of content distributed as well as the user population interested in distributed content are likely to continue to increase at a phenomenal rate.
- Quality of service: While server-side techniques might mostly suffice for ensuring reliable and smooth delivery of traditional unicast streams, more sophisticated solutions are required for uninterrupted delivery of multi-point traffic. In order to achieve scalable content distribution, most proposed schemes rely on organizing servers and/or end-hosts into distribution trees or meshes. As a consequence, the quality and performance of content downloads is dependent on the health of multiple computing and networking elements -- the failure of any one of them could result in degraded service. If these intermediate nodes are servers deployed for the purpose, this raised questions as to which organization is the provider of these servers, and how are they compensated for the service. If end-nodes are used to forward the traffic, then the challenge is providing high quality service especially when there is churn in the end-user population.
- Manageability: Commercial solutions such as Akamai's CDN have demonstrated the feasibility of large-scale infrastructures for content distribution and storage, but they are solutions with service and pricing targeting large corporations. As the vast majority of end-users generating personal content would desire cheap solutions and as the management costs of distribution channels constitute a significant fraction of their operating costs, more study is required in developing self-configuring and self-managing networks.
- Robust incentives: Incentives for the content distributor, network operators, and end-users might not necessarily be aligned to ensure optimal allocation of network resources. For instance, overlays of end-users might form peerings that are suboptimal from an ISP's perspective with the resulting traffic subverting the ISP's traffic engineering policies or incurring it exorbitant peering costs. Faced with increased costs, ISPs might throttle certain kinds of peer-to-peer traffic resulting in abysmal user perceived performance. Further, distribution solutions that rely on end-user's upload contributions (in order to reduce server loads) need to provide proper incentives for the participating users to contribute enough resources to the system. Designing a system that provides a robust set of incentives is thus a challenging issue.

Addressing the above set of research questions requires a distributed infrastructure such as

GENI. A meaningful testbed for evaluating content distribution systems needs to have points of presence at multiple geographical locations and within multiple ISPs, in order to study whether the proposed solution can work in global settings and whether it provides the right incentives to the various players. As some solutions might rely on tight integration with the networking infrastructure, GENI's proposal to have nodes colocated with the Internet backbone routers would also be extremely valuable. Edge diversity, in terms of participating computing devices, their upload/download performance, and their physical connectivity, would also help in identifying solutions that are robust under a wide range of operating conditions.

Naming Systems for the next-generation Internet: Naming systems provide identifiers for components of a system, such as users, Web sites, computers, etc. In the Internet, e-mail addresses (e.g., whitehouse.gov), Web sites names (e.g., google.com), computer names (e.g., mailserver.comcast.com) are all domain names, which are implemented by the Domain Name System (DNS). DNS forms the glue that ties users, Web sites, computers all together to form what the user experiences as the Internet or Word-Wide Web (in fact, most users cannot tell the difference between the two).

Although DNS has been essential to the success of the Internet, the way it is designed has raised both design and implementation problems. For example, there is political fighting about which institution (United Nations, U.S. government, an independent non-profit) should be in charge of the top-level name spaces (i.e., .org, .com, etc.) and the corresponding servers. Because DNS names are used as brand names, there are fights over ownership of names and trademark issues. Adversaries can spoof names, because the base DNS lacks security, while the current security extensions are difficult to deploy because they require a central public-key infrastructure. Because name resolution requires Internet access, two computers disconnected from the Internet cannot use DNS names to communicate, and must have a separate naming plan. DNS also has several implementation problems: the root servers are easy targets for denial-of-service attacks, and name records are not easily updated because they are widely cached. (The alternative of not caching leads to a different problem: sudden demand for name resolution can easily overload name servers.)

These problems have led to much research in naming systems, which has produced fundamental, new techniques for design and implementation. These ideas include semantic free references that have no commercial value by themselves, self-certifying names that are inherently secure, naming that builds on trust relationships in social networks, distributed hash tables to resolve flat names efficiently, etc. There are two projects in this area funded by NSF as part of FIND. One scheme, *Transient Network Architecture*, by Kahn, Jerez, Abdallah, Heileman and Shu, is based on a global name space for objects, where identifiers are globally unique, and are resolved into lower-level details (such as location and current concrete implementation) by means of a distributed but coherently managed identifier resolution service. One of the specific challenges is to design and build a distributed resolution service using peer-to-peer approaches. The other proposal, *User Information Architecture,* by Morris and Kaashoek, similarly assigns identifiers (or names) to digital objects, but starts with independent user-specific name spaces that "get to know each other" in a bottom up manner. The propagation of the information necessary to resolve names is carried out using "gossip" protocols among relevant devices. Security is also achieved "bottom up", with initial exchange of keys through physical interconnection of related devices. The research challenges include routing, security of the namespaces, and schemes for effective name-space management. These two proposals raise very different issues of scale, consistency, usability, management, and resilience. Many of these

questions can be evaluated by the deployment and comparative evaluation of the competing ideas.

To validate whether these ideas work on the scale of the Internet, a testbed is needed that spans the world, different administrative domains and cultures. The experimental research here will consist of the deployment and evaluation of competing approaches to solving these problems. GENI would provide the needed facility to experiment with new naming systems on a large scale and allow the GENI users to live in a different world that isn't necessarily backwards-compatible with the existing Internet. This allows researchers to freely experiment, compare different plans, and answer fundamental questions about which new design is best.

3.1.3 A network for global sensing

If we accept that in 10 years, most of the computers will be small, embedded processors rather than large, powerful processors, then a future Internet should be designed to support the application patterns of these devises. Perhaps the most challenging and important paradigm to support is global sensing, which involves integration and manipulation of data across the world, not in a locale. Sensor networking has demonstrated great potential in many areas of scientific exploration, including environmental, geophysical, medical, and structural monitoring. GENI offers the opportunity to bridge across multiple discrete sensor networks to provide monitoring of physical phenomena at a global scale. In addition, GENI can provide the infrastructure for querying and fusing data across multiple (possibly overlapping) sensor networks in different scientific and administrative domains. A rich application domain for this infrastructure is geophysical monitoring. Examples include monitoring seismic activity along fault lines and at volcanoes, and GPS-based geodesic measurements of plate movements. The NSF EarthScope [EARTH] initiative is building the sensor infrastructure, and tying this source of data into GENI would enable a "continental scale sensor network" supporting a wide range of real-time geophysical monitoring applications.[2]

The availability and cost of sensor hardware, such as CCDs, microphones, motion detectors and temperature sensors, has improved dramatically over the past several years, suggesting that the vision of global sensor networks is within reach. However, while the state of sensor hardware has progressed rapidly, the software infrastructure needed to make these devices useful to applications is still sorely lacking. In designing this infrastructure, the sensor networking research community must overcome a number of challenges. We highlight some of these key challenges in the following paragraphs.

Federated deployment. In order to achieve its global reach, the Internet relies on a federated collection of Internet service providers (ISPs). Similarly, it is unlikely that any sensor infrastructure will achieve the goal of global coverage without the cooperation of many organizations or individuals contributing their sensor data feeds. However, enabling this cooperation requires new protocols to exchange information about available sensor feeds, policies for use of the sensor data, etc. Practical experience with such deployments on GENI will likely provide important guidance in the design of these protocols.

---

[2] This is equally true for other NSF earth observatories such as the National Ecological Observatory Network (NEON), http://www.neoninc.org, and the Ocean Research Interactive Observatory Networks (ORION), http://www.orionprogram.org.

P2P deployment. Allowing individuals to contribute their own sensor observations may play an important role in monitoring the environment. For example, humans may be able to direct their sensing activities (e.g., turning on their cell phone cameras/microphones) to report interesting or desired phenomena. Researchers must develop techniques to ensure that new sensor feeds can be easily added to the overall infrastructure.

Scale. The protocols and systems used in this infrastructure must scale with the number of devices, the number of applications, the number of users and the volume of sensor data collected. Many estimates for the sensor networking market suggest that sensors are likely to outnumber any other form of Internet host by a significant factor.

High bandwidth sensor streams. While most currently deployed sensors produce data at low rates, future deployments may not be so limited. For example, video and acoustic sensors may be an important part of the future infrastructure. In addition, while each sensor may not produce significant amounts of data, the sheer scale of the infrastructure may make the aggregate data rate significant. Researchers must address the issue of effectively managing bandwidth. This may include enabling the infrastructure to perform in-network processing and filtering of sensor data streams.

Privacy. Monitoring the environment raises a number of obvious privacy concerns. Researchers need to provide mechanisms for both observed users as well as sensor data providers to express and enforce their privacy desires. However, it is likely that some of the privacy policies may never be practical and it may be necessary to explore what novel forms of privacy real users may be willing to accept.

Data accuracy/verification. Any application running on this infrastructure will likely incorporate data from a large number sensor data sources. The accuracy of the sources and the trustworthiness of the sensors' owners are likely to vary widely. Researchers must provide some way for sensors to calibrate themselves and for applications to determine how reliable a particular sensor reading is.

Application development tools. A large scale sensor deployment will not be useful unless practical applications can be developed that use the sensor data. However, existing communication primitives are far too low-level and are often not well suited to this application domain. For example, existing naming and service discovery does not match well with the needs of discovering the set of sensor data feeds most appropriate for a particular application. Similarly, existing socket primitives are not well suited to the data transfer requirements for sensor applications. Researchers will need to develop and test a wide range of primitives as new sensor types are deployed and new sensor applications are developed.

To study these sorts of question, it is necessary to prototype this infrastructure on the GENI facility. Specific experiments include developing and testing an appropriate overlay network infrastructure for querying individual sensor data sources [GIB03]; constructing flows of sensor data through multiple stages of filtering, processing, correlation, and aggregation [PIE06]; and delivering the resulting data to the end user. There are two funded FIND proposals in this area: *Sensor-Internet Sharing and Search*, by Heidemann, Cho and Hansen, and *Network Fabric for Personal, Social and Urban Sensing Applications,* by Srivastava, Burke, Estrin, Hansen and Paxson.

3.1.4 An architecture for relayed communication

Both of the previous ideas involve communication patterns that are not interactive end-to-end,

but which proceed by stages, where information is positioned for rapid delivery, integrated, and then forwarded. One view is that this general paradigm may come to dominate the future Internet. (Even for telephony, communication is often "not interactive", a phenomenon called "phone tag".) One of the drivers of this vision is the revolution in wireless access technology, which alters dramatically the nature of Internet traffic and challenges the basic assumptions upon which its protocols were built. Where the end-points of Internet traffic were once stable and predictable, they are increasingly embodied in wireless devices, whose numbers and information rates are increasing dramatically, and which have left the stable environment of the home and office to wander in the mobile world. They have introduced instability to Internet connectivity and made the easy assumptions of end-to-end traffic flow increasingly untenable. Because the changes caused by wireless mobility are fundamental and pervasive, their solution requires comparably fundamental changes in the architecture and protocols of the future Internet.

One such architectural approach, Delay Tolerant Networking (DTN)[FAL03], approaches this problem by taking advantage of storage available in the network to help overcome link disruption. It also provides the beginnings of a standardized approach to constructing interoperable proxies using a general naming scheme. Such proxies have been used extensively for interconnecting "radically heterogeneous" networks, such as sensor networks, with the Internet. DTN is also being pursued by DARPA for use in military tactical networks that may suffer from unplanned disruption [DTN04].

DTN and related concepts have been under development for a number of years, and they are obvious candidates to deploy and test on GENI. There are two FIND proposals funded in this area. One of them is motivated by a particular usage scenario, disaster recovery: *The Day-After Networks: A First-Response Edge-Network for Disaster Relief,* Luo, Abdeizahar and Kravets. This proposal exploits staged, opportunistic delivery to take advantage of the intermittent and highly variable networks that might first be available after a disaster. In particular, their architecture involves service level forwarding (rather than host-to-host packet level forwarding), and uses role-based anycast as the prevalent delivery mode. They implement this through opportunistic exploitation of heterogeneous technology, and careful use of service-level message flooding. Their proposal contemplates a complete architecture, which means they will have to address a range of requirements, including manageability, security, and incentives.

The other proposal in this group uses staged delivery to deal with the intermittent nature of many wireless devices; *Postcards from the Edge: A Cache-and-Forward Architecture for the Future Internet,* Yates, Paul, Raychaudhuri and Kurose. In contrast to the proposal by Luo *et al.*, this architecture makes use of well-known and stable intermediate nodes called *post offices*, the mode of delivery is to a named destination, not anycast to a role, and the topology of the core of the net is planned and managed. However, the intermittent nature of the wireless and mobile edge nodes is very similar to the first proposal. Research challenges include the design of a naming protocol and a routing protocol.

3.1.5 A scheme for universal mobility

Section 2.1.2 discussed the implications of emerging wireless technology, and the need to plan for ubiquitous mobility. Unfortunately, today's Internet does not support mobility well. We identify two problems (at a minimum) that must be rectified to meet the needs of mobility: maintaining connectivity, and achieving effective transport.

Connectivity: As many researchers have noted in the past, a fundamental problem is that the Internet uses IP addresses to combine the notion of unique host identifier with host location. For a mobile host to have seamless connectivity and continuous reachability, it must retain its identity while changing its location.

Previous mobility proposals decouple this binding by introducing a fixed indirection point (e.g., Mobile IP), redirecting through the DNS (e.g., TCP Migrate), adding unique identifiers to hosts (e.g., Host Identity Protocol), and using indirection at the link layer (e.g., cellular mobility schemes). However, none of these schemes appear to offer a complete solution.

Here is one set of desiderata for properties in order to fully realize the promise of ubiquitous mobility:

(1) Efficient routing: packets should be routed on paths with latency close to the shortest path provided by IP routing.

(2) Efficient handoff: the loss of packets during handoff should be minimized and avoided, if possible.

(3) Location privacy: the host's topological location should not be revealed to other end-hosts.

(4) Simultaneous mobility: end hosts should be able to move simultaneously without breaking an ongoing session between them.

(5) Personal/session mobility: a user should be able to redirect a new session or migrate an active one from one application or device to another one when a better choice becomes available.

One recent design that achieves all these properties is based on the Internet Indirection Infrastructure (i3) architecture [STO02]. The i3 scheme is fairly mature, and we describe it in some detail as an example of a concept that might be tested on the GENI facility.

Unlike IP, with i3, each packet is sent to an identifier, not to an address. To receive a packet, a receiver creates a *trigger*, which is then stored at an i3 node. The trigger is an association between the packet's identifier and the receiver's address. Each packet is routed through the i3 infrastructure until it reaches the i3 node that stores the trigger. Once the matching trigger is found the packet is forwarded to the address specified by the trigger. Thus, the trigger plays the role of an indirection point that relays packets from the sender to the receiver.

The particular design of i3 triggers makes it well-suited to support mobility. A mobile host that changes its address from R to R' as a result of moving from one subnetwork to another can preserve the end-to-end connectivity by simply updating each of its existing triggers from (id, R) to (id, R'). To achieve efficient routing, end-hosts can choose triggers that map on nearby i3 node, since end-hosts can dynamically change triggers without disrupting end-to-end connectivity. This drastically alleviates the triangular routing problem, as packets need not travel to nodes far away from both the sender and the receiver. Since triggers are periodically refreshed, end-to-end connectivity recovers gracefully from node failure. If an i3 node fails, the triggers stored at that node are inserted at another i3 node next time they are refreshed. Furthermore, replicating triggers at the i3 level, or using backup triggers can make i3 node failure completely transparent to end-hosts. This is in contrast with mobile IP, where the home agent failure will sever all the connections of the mobile host.

The fact that an i3 host sends packets to an identifier rather than an IP address provides both location privacy and support of simultaneous mobility. The receiver can trade routing efficiency for privacy by placing the identifier anywhere in the network. Furthermore, the fact that the receiver does not need to update the trigger identifier when it moves (it needs only to update its IP address in the trigger) makes the receiver mobility transparent to the receiver. This allows both end-hosts to move simultaneously without breaking the connection between them.

The i3 scheme is fairly mature and has been tested on PlanetLab. Unfortunately, the size and resource scarcity of PlanetLab, both in terms of bandwidth and CPU, makes it very hard, if not impossible, to achieve critical mass. To provide seamless mobility, every packet must be forwarded through the i3 infrastructure, and PlanetLab makes the forwarding overhead unacceptable. GENI would represent an ideal platform to test and evaluate the mobility service over i3. The scale of GENI would allow us to understand how the design behaves at large scale, as well as the robustness properties of the design in the presence of realistic failures and network congestion. More importantly, the GENI wireless testbed would allow one to extend the design to seamlessly operate across heterogeneous link layer technologies, and experiment with new link layer protocols. Finally, the virtualizable routing platform would support i3 at very high speeds, which will significantly improve the scale and performance, and ultimately increase the user base.

Transport: The second set of problems arises from the variable and often poor connectivity that mobile hosts and networks endure. The ubiquitous TCP/IP protocol used for most Internet services has several known weaknesses when applied to mobile data scenarios. In the extreme, mobile nodes are sometimes disconnected, and the IP network layer framework does not support disconnected operation or caching/storage within the network. Section 3.1.4 discusses solutions to this problem. Even when there is connectivity between end points, variable throughput and error rates may contribute to poor performance. For example, the window flow-control mechanism in the TCP transport layer performs poorly over wireless access links with high error rates. Numerous solutions to these problems have been investigated by the wireless networking research community, including mobility service overlays and modified TCP or all-new transport layer protocols, but none of these solutions have migrated to general use due to legacy staying power and the difficulty faced by innovators in deploying their protocols on a large-scale network to test performance and end-user acceptance.

A GENI experiment of near-term interest to the mobile networking community would be to deploy one or more alternative protocol solutions on an end-to-end basis with a significant user population. Experimental measurements of interest include short-term numerical performance measures such as packet delay, packet loss rate and user throughput for specified applications and mobility patterns. In addition, longer-term service quality measures such as the percentage of dropped connections and level of availability will be measured.

3.1.6 Reliable communication with tight time bounds

In contrast to schemes for staged delivery, there is an objective that a future Internet should support the option of bounded-delay real time interaction for such purposes as remote control, telephony and real time streaming, and so on. While there has been substantial work on how to combine the current best-effort traffic delivery model of the Internet with services that provide delivery with tight bounds on delay, there have been no large-scale, wide area demonstrations of these integrated schemes. There is much uncertainty and disagreement as to whether these schemes can provide integrated, multi-service traffic delivery in a cost-effective way. An

infrastructure that cleanly and reliably supported these technologies could drastically change the way we communicate. To make progress in this area, we must identify and characterize timing requirements for media-centric application and remote control applications (e.g. robotics and other mechanical systems); we must agree on the technical parameters of "real time" (low average latency, bounded maximum latency, low variance, etc.), we must select algorithms for allocation and scheduling of capacity to meet these requirements, and we must perform a large-scale evaluation and demonstration of these ideas on GENI, including support for critical applications, such as low-latency and error-free delivery of high-quality imagery, and low-latency control of a remote robotic device. The specific experimental objectives are both to support these application classes, and to demonstrate that we can use the same network resources to support other service classes at the same time.

These sorts of real-time applications often also have high requirements for reliability, resilience and availability. The example often used to illustrate the suite of requirements is remote surgery. This additional set of requirements implies that we must augment the allocation and scheduling mechanisms with schemes that provide for backup capacity over disjoint routes, rapid failover from one path to another (or perhaps even simultaneous transmission along these disjoint paths), and so on. Section 3.1.7 describes possible approaches to building a more robust, resilient network, which is an essential part of supporting high-demand applications. While there has been past research on "real time" mechanisms, there has been no large-scale demonstration of a complete set of mechanisms that together provide highly-reliable, bounded latency real-time interaction. The demonstration of such a scheme is a major objective for a future Internet.

3.1.7 An architecture for a secure and robust Internet

Most of the previous experiments and demonstrations presume that there is a framework for secure and robust provision of network service. Section 2.1.1 discussed the critical importance of this capability.

Altering the current state of affairs will require a sea change in computer systems and networks. First, we need security architected from the ground up, to support a unified and reasoned framework for enforcing security policy. Today we have a collection of mechanisms and schemes, but no architecture. Second, we need abstractions and metaphors that enable users to better understand how to specify and interpret policy in this framework, so that users can specify policy at coarser levels of detail (e.g., what tasks should be allowed, not what files can be accessed in what ways) and with better comprehension of the results.

A design for security must be holistic, and deal with issues at all layers. Securing a single layer is insufficient, as an attacker can exploit vulnerabilities in the layers that remain unsecured. To briefly illustrate this problem, consider the issue of BGP routing security. BGP is a routing protocol used to mediate most packets that traverse the public Internet. To secure BGP, researchers have proposed securing the communication between two BGP speakers by having them share a secret key and adding key-dependent MD5 checksums to each packet exchanged. Unfortunately, this does not secure the semantics of the exchanged information, and a malicious operator (or equivalently, a corrupted router) with appropriate credentials can still send routing messages that disrupt communication system-wide. Even if we secure the semantics of the routing information, the routing protocols can be disrupted by exploiting the vulnerability of the TCP connection carrying the BGP packets to denial of service attacks.

Thus, only a thorough, system-wide approach to security is viable, addressing naming, routing, connection management, resource allocation/denial of service, network management, and so forth. While many researchers have begun to tackle these issues, to be practical we need to understand the relationship of these various technologies with each other and collectively on system-wide security. Further, if these technologies have any hope of being widely adopted, we must also demonstrate that they can achieve system-level security at a practical cost. It is this last point where GENI is essential, as a platform where a new secure networking architecture, providing strong assurance of communication availability even under attack, can be demonstrated to work in practice on a national scale network connecting millions of users, at Internet speeds and reasonable cost. Section 3.2.5 catalogs a number of specific security experiments and demonstrations that we contemplate performing using GENI.

Resilience and robust operation: Although not perfect, the telecommunications industry does a reasonable job of building dependable telephone systems. The size of the software used to control a telephone switch is on the order of 10 million lines of code. This software, coupled with the underlying hardware, is designed to have (and routinely achieves) no more than *three minutes* of downtime *per year* from all causes. These impressive numbers are achieved using a variety of hardware-redundancy techniques, self-checking software, and a rigorous development process. Unfortunately, these dependability-enhancing techniques are primarily geared towards software that is developed by a single organization and runs on a single computer. A system such as the Internet, whose components run on a geographically distributed set of hosts and are written by multiple organizations, requires new techniques to achieve dependable operation.

Despite decades of effort to build perfect hardware and software, hardware failures, software bugs, operator errors, and malicious attacks continue to cause failures in computer systems. Such problems limit our ability to use the Internet as a critical infrastructure and to deploy new services in or on it: for example, the Internet can be destabilized by compromising a single router or a modest number of end-hosts, web services are susceptible to attacks from determined hackers as well as compromised machines of innocent users, and proposals to extend the Internet to provide more sophisticated services like mobility support only broaden the risks. A fundamental challenge for computing in general and networked services in particular is "How do we construct reliable systems from unreliable components?"

One approach to addressing this challenge is a body of work that has developed the theory and basic practice of Byzantine fault tolerance (BFT). A BFT system uses redundancy to mask faults and provide correct operation even if some system components malfunction in arbitrary ways or are controlled by malicious parties. As evidence mounts that simple "fail-stop" failures account for only a fraction of real-world failures and as falling hardware costs make replication an attractive approach, BFT is an increasingly attractive technology.

Although the BFT paradigm has seen an extensive body of theoretical and systems research over almost three decades, we have little experience with large-scale deployment. GENI offers the opportunity to develop and deploy BFT-based systems, which will help answer questions like: (1) What are the right abstractions for BFT services? The BFT theory has been well developed for primitives like agreement protocols and quorums, but higher level abstractions like locks and reliable databases may be easier for programmers to use. (2) How much reliability will BFT systems deliver in reality? The BFT model assumes a bound on the number of node failures, but in practice, failures are often correlated; real-world deployments offer a

way to understand the extent to which correlated failures can be managed in practical settings. (3) How to model real-world failures? The BFT model allows for arbitrary failures by some subset of components. This flexibility is at the root of BFT's power, but it also increases protocol costs and limits deployment of the approach to environments where a small fraction of nodes can fail simultaneously. Extending BFT to better model real-world could reduce replication costs or extend the environments for which replication is useful.

GENI can help bridge this gap between theory and practice by providing a testbed to deploy and validate more practical approaches for BFT. A massive-scale distributed edge testbed comparable to large content distribution networks today can provide the necessary impetus for this. Augmenting a network testbed to support realistic edge services allows research in networking to go hand in hand with research in distributed services run on top these networks.

Below, we describe some concrete technical challenges that such a testbed can help address.

1. **Asynchrony:** Traditionally, distributed systems abstract the underlying network by assuming that communication is either synchronous or asynchronous. Because real networks are failure-prone and may be unavailable for long periods, dependable systems must assume asynchronous communication. Unfortunately, operating over an asynchronous network increases replication costs and weakens the guarantees that replication can provide. For example, if an infrastructure for near-perfect end-host failure detection could be developed, distributed systems could continue to make progress even when more than half the replicas are unavailable.

   GENI can help develop real networks that better match desirable theoretical models. As a first example, it could enable researchers to better understand how to build a synchronous reliable network, i.e., one that guarantees message delivery within a finite time bound. Although link and network layer technologies to provide delay guarantees exist, today's best-effort IP networks do not expose this ability to end-systems. Second, it could enable networking and distributed systems researchers to work hand-in-hand to develop better failure detectors. Although there is an extensively developed theory of failure detectors, developing practical and useful failure detectors with network support is a challenge.

2. **Data Centers:** The rise of massive data centers often hosting third party content makes BFT crucial. Although researchers have been developing increasingly optimized approaches for practical BFT, the data center industry has not adopted these ideas. GENI can help make research more practical by providing large edge clusters that can host real Web services like content distribution or multi-tier Web services. The prominence of virtual (or physical) machines as a resource provisioning mechanism in data centers is already leading to Web service tiers being designed with replication in mind. Given that service tiers are replication aware, incorporating BFT seems like a natural next step. A GENI edge cluster can provide the necessary incentive to spur adoption of the BFT paradigm by real services as well as enable researchers to develop better fault models and tolerance approaches based on this experience.

3. **Correlated Failures:** The BFT model implicitly assumes that nodes fail independently and that the number of failures is bounded. For replicated distributed services, failure independence may be achieved by N-version programming. However, network or routing failures are often correlated in unpredictable ways. GENI envisions the use of virtual

machines inside routers to enable concurrent competing architectures that could provide the much needed independence of failures to build a robust routing infrastructure.

4. **Disconnected Operation:** Integrating support for disconnected operation is crucial for at least two reasons. First, real networks will be failure-prone and will have imperfect coverage in the foreseeable future. Second, the increasing use of hand-held devices that can communicate in ad-hoc mode (e.g., Microsoft's music player Zune, or OLPC laptops) offers an opportunistic moderate-bandwidth high-delay network that can be fruitfully used by delay-tolerant applications. Recent advances in distributed systems research such as PRACTI replication [BEL06] provide the theory for enabling delay-tolerant applications across heterogeneous devices, consistency requirements, and opportunistic transfers. Making such systems Byzantine fault-tolerant is critical given the inherently untrusted transport infrastructure.

   GENI's support for delay-tolerant edge network testbeds can help in two ways. First, it enables researchers to translate theory to practice and vice-versa. For example, it is clear that PRACTI needs to be Byzantine fault-tolerant to be practical, but it is difficult to speculate about realistic fault models, especially those involving user or operator errors, without the opportunity to deploy these applications for real. Second, although the field of delay-tolerant or disruption-tolerant networking (DTN) is growing, there is little exchange of ideas between networking researchers and distributed systems researchers even though they are targeting the same environment. One reason is the lack of real and controlled testbeds that allows researchers to study what the DTN network stack should look like all the way from the link layer to the application layer, and comparing this with the traditional IP-based network stack.

5. **Rational Behavior:** BFT is too strong a model in a world where most nodes are "selfish", but only a few are Byzantine. Recent efforts such as the BAR model pave the way to building distributed systems that tolerate a mix of Byzantine as well as rational nodes. Although initial efforts have demonstrated benefits of distinguishing between Byzantine and rational nodes for distributed systems, there is value to exploring a similar approach for the underlying network infrastructure. For example, address hijacking and inserting false routing advertisements are examples of Byzantine behavior in interdomain routing, but holding back (or not holding back) a routing update long enough for selfish benefit is acceptable rational behavior. The interaction of Byzantine and rational behavior is further complicated by the possibility of collusion; analyzing the effects of collusion is difficult even in traditional game-theoretic models. GENI can help by enabling a controlled network testbed to deploy and validate routing architectures as well as distributed systems based on the BAR model.

6. **Secure Identities:** The BFT model implicitly assumes strong verifiable identities. Otherwise, faulty nodes can create an arbitrary number of identities violating the bound on the maximum number of failures, an attack referred to as the Sybil attack. Conversely, trustworthy identities are envisioned as a means to improve security on the next generation Internet. GENI will help us understand how to develop an infrastructure for secure identities that can be used for a broad range of network as well as end-system services.

The GENI facility will enable researchers to experiment with new techniques for building dependable distributed systems. For example, GENI will facilitate experimentation with new communications protocols that could allow components of a large distributed system to

exchange information reliably and securely. Because GENI nodes will have significant computational and storage capabilities, experimental validation of new dependability techniques that rely on large-scale replication of code and data can also be performed. The sensor networks attached to some GENI nodes will permit these kinds of experiments to be expanded to address systems that touch the physical world.

3.2 Building blocks for a future Internet

Another way to contemplate what a new Internet might look like is to catalog some of the key components of the current Internet, note what is wrong with each part, and list some of the proposals that have been put forward to improve them. This approach has the risk that it can lock us too much into the current set of parts, but it has the merit that it permits a concrete example of what specific experiments on GENI might look like. So with that warning, we can look at alternatives to mechanisms found in the current Internet.

3.2.1 Packets and multiplexing

A basic assumption of the Internet is that data is broken into packets, which are then multiplexed statistically along communications paths. The decision of how to multiplex an individual packet is made based on information in the IP header of the packet – the address and type of service bits. Most (though not all) researchers conclude that the concept of packets is a good one that should be a part of a future Internet. But in the center of the network, there is an increasing view that the information in the IP header is too fine-grained, and that the multiplexing decision should be based on some aggregation of the IP header information. Today, this is done outside the architecture, using a separate mechanism (such as MPLS[LEF02]). If **routing and management of aggregates** were included into the architecture of the Internet, it would allow both packets and aggregates of packets to be handled in a unified way. In particular, the concepts of routing, traffic engineering and topology management should be unified in a future Internet. Fault recovery should be unified across layers. The inclusion of switched optical components in GENI will allow researchers to experiment with algorithms for rapid reconfiguration of aggregates.

While statistical multiplexing of paths leads to good link utilization and cost-effective design, it is also a security risk, in that an attacker may be able to flood certain links to the point where good users are squeezed out. There are several approaches that have been proposed to solve this problem. One is **Quality of Service**, which is now being used in private networks, but only partially in the public Internet. Another approach is **virtualized resources**, in which simple statistical multiplexing is replaced with a more complex layered approach to sharing in which classes of users or activities are given static shares, and only within these classes is there dynamic sharing. GENI will be used to test the concept of virtualization. Another approach to controlling abuse is **diffusion routing**, discussed below.

3.2.2 Addressing and forwarding

Once we agree that the network will carry packets, the next step is to design the mechanism that allows packets to be forwarded across the network. The Internet contains elements called routers, which look at the *address* in packets to determine how to forward them. The original Internet assigned a global address to every destination, and allowed any computer to send a packet to any place. This open pattern of communication was critical to the early success of the Internet, but has caused a number of serious problems, which only became apparent over time. For this one topic of packet addressing and forwarding, we have cataloged over 24 proposals

for alternative addressing and forwarding schemes, most of which have gone nowhere, because there is no way to validate them. GENI will allow us to try out alternatives to today's scheme that might provide better security, better management, and better functionality.

One problem with global addressing is that it allowed the Internet to be a vector to deliver security attacks, since any machine, including a malicious one, can send traffic to an attack target. A future Internet must provide what has been called **trust-modulated transparency**: trusting nodes should be able to communicate at will, as in the original conception of the Internet, but nodes should be protected from nodes they do not want to communicate with [CLA03]. There are several approaches to achieving this balance, which we expect to validate using GENI. One is **address indirection**, in which senders do not know the address of the receiver, but only a name. An example of this scheme is i3 [STO02], described in Section 3.1.5. A protected element in the network (which would have to be invented for the purpose) would check the identity of the sender, and decide whether to forward the packet to the address of the named recipient. A second approach is the **permit** approach, in which the receiver gives to the sender a special token, which is then included in the packets from the sender. Again, a protected node in the network would check the token to decide whether to forward the packet [AND03b]. These schemes, in general, are examples of taking the concept of a **firewall**, which is an afterthought in the current design, and considering from scratch how to integrate this component into the Internet as a first-class element.

Minimizing Packet Buffers

All Internet routers contain buffers to hold packets during times of congestion so that the network can accommodate transient bursts without incurring losses. Buffers are what allow packet-switched networks to simultaneously achieve high-quality service and high utilization. Given the ubiquity and significance of packet buffers, one might expect buffer sizing to be thoroughly understood, based on well-grounded theory and supported by extensive experiments. Instead, for many years buffer sizing was based on a widespread but unsubstantiated rule-of-thumb that routers should provide at least one round trip time's worth of buffering, often assumed to be 250ms.

A 2004 paper [APP04] challenged this conventional wisdom, arguing (using analysis, simulation and some anecdotal experiments) that buffers can be greatly reduced in backbone routers because of the large number of multiplexed flows. Specifically, [APP04] argues that the buffer size can be reduced by a factor of $\sqrt{N}$ (where N is the number of flows) without reducing link utilization; moreover, these smaller buffers would, during periods of congestion, significantly reduce maximum jitter and end-to-end delay. These results suggest that users would get a significant reduction in latency and jitter, without the operators giving up utilization of their networks.

Today, a 10Gb/s packet buffer holds about 1,000,000 packets; the results above suggest they can be reduced to 10,000 packets. Recently, it's been proposed that packet buffers in backbone routers could be reduced still further - to as small as 20-50 packets [ENA06, WIS05] . These are more radical proposals, and come at the expense of some link utilization, on the order of 10-15%. These results might be applicable to all-optical routers, for which recent integrated optical buffers have been built that are capable of holding a few packets.

These results have the potential of changing the way commercial switches and routers are designed and deployed. Backbone routers are generally limited by power consumption; on some linecards the buffer memory consumes a third of the power and a third of the board-space. In some commercial switches the buffer memory is more than 25% of the component cost. Thus, smaller buffers could lead to significantly simpler and cheaper switches and routers.

However, several authors advise caution, arguing that oscillations and packet loss can occur with very small buffers [DOV05]. The truth is that no one knows for sure what will happen in a real network, as all of the experimental results to date are quite anecdotal, and limited to single links, small networks and lab experiments. Thus, more experiments are needed before they will have the credibility to lead to a widespread reduction in buffer size.

And herein lies the problem. It is not possible to just measure buffer occupancies in today's networks; to test these hypotheses requires reducing buffers in the routers by factors of 10, or 10,000, and then running the network for long periods of time to find out if the hypotheses hold under a broad set of conditions. How could a responsible network operator take the risk of disabling most of the buffers and potentially disrupting their customers' traffic? And even if they were willing, it turns out that one can't set the buffer size accurately in commercially deployed routers, and none measure buffer occupancy in real-time.

In the absence of a realistic, programmable network, researchers have resorted to building their own switches and routers, where buffers can be flexibly controlled and accurately measured, and then constructing a small lab network. While this provides some relevant evidence, it won't come close to passing the credibility test for a network operator, or an equipment vendor. Router vendors have a vested interest in keeping buffers large as it helps justify a much bigger margin than for - otherwise almost identical - Ethernet switches.

Given the impact this small-buffer hypothesis might have on the performance of the Internet and the design of routers, it seems crucial that we evaluate it more thoroughly. To do so will require: (1) routers for which we can accurately set the size of the buffer, and measure the occupancy in real time, (2) a network built from these routers (natively over links, not as an overlay, as buffer occupancy depends critically on link delays), and (3) real user traffic that will allow the hypotheses to be tested with lots of users, and lots of applications, over long periods of time. GENI naturally supports all these requirements.

A second problem with the original addressing scheme is that it did not take into account mobile devices, which are becoming the norm, and may dominate the Internet in 10 years. Today, Internet addresses are used to convey a weak form of identity as well as location on the net. Since the IP address captures the notion of identity, it is not possible to change the IP address in an ongoing conversation, which means that a node that is mobile cannot change its address as it changes it location. A future Internet must have a scheme for **dynamic address reassignment** and a scheme (or several) for automatic **connection persistence** for mobile end nodes, for example, [MOS06].

On the other hand, as we better deal with mobility, the value of an address as a signal of identity may erode. This raises the question of whether there needs to be some explicit form of identity that is visible to an observer of packets in the network. Different answers have very different implications for privacy, for accountability and for policing. One response to this question is that there will be different needs for identity in different parts of the network, so the packet header should include an identity field, but not a rigid specification of what that field should contain. One proposition for an experiment on GENI is a **semantics-free identity field** in the packet header.

Today, the Internet names services (such as Web or email) using "well-known ports" — numerical indices that are statically assigned to each application at design time. Since these port numbers are included in each packet, this permits any observer in the network to determine what application is being used. And since these numbers are statically assigned, an attacker can easily launch an attack against an application on a given host, just by combining a host address with the port number, and using that destination as the target of an attack. An alternative would be to design a new mechanism for **service rendezvous**, and to use random port numbers to identify connections. This change, combined with an increase in the range of port numbers, would essentially eliminate the value of the attack known as port-scanning, and would provide more privacy from observers in the network. However, a sparse port-space would change the whole security landscape by changing what firewalls can do based on packet inspection. In fact, this change would force a complete re-conception of what a firewall does and the balance of power in the space of attack and defense. The alternative would also change the economic landscape by making it harder for Internet Service Providers to discriminate among customers based on what applications they want to run. Presumably, they would respond by inventing some other form of discrimination. The change would make the network more useful to consumers, by eliminating some of the restrictions that are imposed by the invention of Network Address Translation [SRI01] units as network attachment devices.

The design of the original naming mechanism of the Internet, the Domain Name System (DNS), was likewise predicated on open, global addresses [MOC87]. DNS gives an answer to any query, without knowing which device initiated the query or the reason for the query. In a trust-modulated Internet, the naming system may wish to know who is requesting information, and for what purpose, before providing that information. This suggests at a minimum a **semantically richer form of address resolution**, and perhaps even a multi-part negotiation more akin to signaling protocols used for voice calls. This leads to fundamental questions that can be tested on GENI: What is the appropriate division of functionality between naming and network addressing? What, if any, role should out-of-band signaling play in a future Internet? Should network addressing be eroded to the point where a naming/signaling system, rather than a global Internet address in every packet, is the unifying characteristic of the Internet?

A final example of a problem with the current Internet addressing scheme is that IP addresses are normally bound to specific physical machines, but in many cases a message needs to be sent to a more abstract entity—a *service* rather than a *machine.* A scheme called **anycast** has been proposed to solve this problem; this scheme allows the same address to be assigned to multiple machines, and the particular machine to receive the packet is determined by the routing protocol at run time [PAR93]. Anycasting may solve a number of security problems as well as problems of service location and session initiation, but it has scalability and deployment problems [BAL05], and has never been fully elaborated or tested. A new Internet may contain a mechanism like this, which will have to be evaluated on GENI.

3.2.3 Routing

Routing is the process of computing the best path from sources to destinations. Routing is not the same as forwarding—routing computes the best paths, forwarding uses the results computed by routing to take the correct action as each packet arrives at the router.

Today, the Internet uses a two-level routing scheme, with a top-level mechanism called Border Gateway Protocol, or BGP[REK95], to connect different administrative regions, and a second level of protocol inside each region. The **region structure** of the Internet seems fundamental, and in fact may be more explicitly expressed in a future Internet design. This means that we will have to set up experiments on GENI to capture the idea that different parts of the Internet are run by different organizations.

The BGP of today is flawed: it limits the business relationships that ISPs can negotiate[GOV99], it recovers from some failures much too slowly[LAB00], it is not sufficiently secure[MUR06], and under some circumstances it can be unstable and lead to routing oscillations[VAR00]. None of these issues were fully understood until the protocol was put into use on a large scale Internet. Alternatives to BGP are being developed that provide better **convergence after equipment failures**[PEI05]. A platform such as GENI is critical to testing these schemes. Evaluating a route-computation service in GENI would enable experiments that measure routing-protocol convergence delay and the effects on end-to-end performance when topology changes occur. This would involve "injecting" link failures under today's Internet routing architecture and under the new design. This experiment would be difficult to conduct without GENI because simulations do not accurately capture the overheads and delays on the routing software, and operational networks would not permit researchers to intentionally inject failures.

Current work in this area include the FIND project titled *Post-Modern Internet Architecture,* by Calvert, Griffioen, Spring, Bhattacharjee and Sterbenz, which proposes a new network architecture based on the idea that the Internet should provide a set of explicit mechanisms to support (enforce) the various routing policies of stakeholders, i.e. service providers and users. In particular, the design provides a simple packet-forwarding infrastructure, which can be used with a variety of routing (pathfinding) mechanisms. The approach maximizes the separation between forwarding and path selection so that tradeoffs are exposed and optimizations in each dimension can be exploited. One goal of this approach is to clearly separate the policy issues that arise between users and providers as to how traffic is routed. In this respect, the proposed work addresses issues of economics and incentives.

Today, the user has little choice over the route his packets take. There is no equivalent of "picking your long-distance provider" in the Internet, and little support for a host or edge network that wants to have multiple paths into the network. This lack of support for multi-

homing is a major contributor to poor availability. It has been proposed that Internet routing should be redone to support **end-node route selection**[CHI, EST92, YAN04] so that the user has more control over how his packets are carried, both to support multi-homing and to impose the discipline of competition on the service providers. The NSF FIND focus area includes a project titled *An Internet Architecture for User-Controlled Routes*, by Yang, which proposes to study the stability of these sorts of schemes.

A common way of accomplishing route selection today is to use a multi-homed NAT device – by translating into the address associated with the chosen access link, the NAT device can control the direction taken for both outgoing and incoming packets. This is a crude and limited form of address indirection mentioned above. Researchers have proposed the use of an overlay network, which effectively tunnels a packet to one provider over another. Neither approach is integrated into the basic Internet architecture[AKE04]. We need to experiment with this to see if we can design tools that make end-node route selection practical and usable, and to see if this approach actually improves availability, since the approach solves some problems and raises others.

The term "tunneling" describes a class of schemes in which a set of users override the default routing of the network. They do this by employing intermediate nodes in the network, and sending the traffic from source to destination via this intermediate. The actual packets being sent can be encrypted if desired, so that all that can be seen if the packets are inspected is the encrypted data flowing through this intermediate. So the actual traffic is "tunneled" inside the traffic via the intermediate. A tunneling solution to the route selection problem, if cleanly integrated into the Internet architecture, could solve a range of problems. Tunnels are used today for many different purposes: to extend PPP sessions (L2TP and PPTP), to provide host mobility (Mobile IP), to securely transport packets across networks (IPSec), to carry IPv6 over IPv4 (and vice versa), to carry IP multicast traffic over non-multicast routers (mbone), to support VPNs (MPLS, GRE, and IPSec), to shunt DoS traffic (MPLS), to support site multi-homing (GRE), and even to support WAN virtual links (Ethernet-over-IP). All of these are ad hoc point solutions to specific problems, and don't fit neatly into the Internet architecture. GENI can be used to experiment with the use of **tunneling** as a first-class component of the architecture[RAT05], leading to solutions for route selection, traffic engineering, mobility, diffusion routing, and fast failure recovery.

Routing algorithms today attempt to find the optimal path for traffic, given the overall traffic pattern. As the traffic pattern changes, routes must be constantly recomputed. An alternate idea is to take traffic and diffuse it across all possible paths from source to destination. It can be shown that **traffic diffusion** [ZHA05] provides stable traffic allocation to links for all feasible traffic patterns. In other words, it eliminates the need for traffic engineering. It also may improve security by eliminating the ability of an attacker to concentrate his traffic onto one circuit in order to overload it. In order to test this idea, what is needed is a network with a high degree of route diversity, which GENI can provide by means of virtualization.

In today's Internet, the route computation is performed in the same physical devices (the routers) that also forward packets. One proposal for a future Internet moves the route computation out of the individual routers, and into a separate **route computation service**[CAE05]. This approach offers great advantages in consistency, manageability, and scale. It allows competing route computation services to offer alternative algorithms for different customers. However, it raises new challenges for robustness and resilience in the face

of arbitrary failure. This breaking up of the router function will also shift the industry landscape and create new opportunities for innovation and competition. We need to experiment with this sort of scheme in a real network with rich connectivity and real-world failure modes. In particular, since GENI provides the option of interconnection with operational ISPs, it can be used to test new routing regimes in the real world.

A Data-Oriented Internet

The first Internet applications, such as file transfer and remote login, focused strictly on host-to-host communication: The user explicitly directed the source to communicate with another host, and the network's only role was to carry packets to the destination address listed in the packet header. The Internet architecture was built around this host-to-host model and, as a result, the architecture is well suited for communication between pairs of well-known and stationary hosts. Today, however, the vast majority of Internet usage is data retrieval and service access, where the user cares about content but is oblivious to its location. The current architecture can support this service, but it is far from a comfortable fit. For instance, efficient large-scale data distribution requires a set of ingenious DNS hacks, as pioneered by Akamai, and a substantial dedicated infrastructure.

These difficulties can be traced to weaknesses in the Internet's domain naming system (DNS). DNS name resolution is a fundamental part of today's Internet, underlying almost all Internet usage. However, DNS was developed rather late in the Internet's evolution, after many basic pieces of the architecture were in place. For instance, TCP sessions were already bound to IP addresses and the Berkeley Socket API referred to addresses, not names; frozen design decisions such as these limited the extent to which DNS names (or any other naming system) could permeate the architecture. As a result, the current role of naming in the architecture is more an accident of history than the result of principled architectural design.

Some researchers are now taking a "clean-slate" look at naming and name resolution. From a user's perspective, some of the goals of naming are:

- Persistence: once given a name for some data or service, a user would like that name to remain valid forever. There should be no equivalent of today's "broken links" when data is moved to another site.
- Authenticity: users should be able to verify that their data came from the appropriate source, and should be able to do so without relying on third-parties or other Internet infrastructure.
- Availability: data and services should have high availability, in terms of both reliability and low-latency. Such availability is usually provided by replication at endpoints, and the network's role is to route user requests to nearby copies.

None of these goals are achieved by DNS, but they are easily within reach of a clean-slate design. In particular, if the names are self-certifying (that is, if an object's name is based on the hash of the object's owner's public key), then if the owner cryptographically signs the object the user can be assured of its authenticity. Note that this does not involve PKIs or any other form of third-party certification. Such cryptographically derived names are flat (i.e., without semantic content), so they remain valid as an object moves between domains. In this way, flat self-certifying names can achieve both authenticity and persistence.

Thus, the remaining challenge is availability, which must be achieved through the process of name resolution. DNS is based on *lookup*; clients submit names to the DNS infrastructure, which then returns with an address that the client can use to contact the intended target. However, to achieve availability, name resolution should guide requests to a close-by copy while avoiding failures. Doing so in a lookup-based system requires knowing the location of all copies and that of the client, both of which must be learned through ad hoc mechanisms. An alternative approach being contemplated in these clean-slate designs is *name-based routing*. Here, the name resolvers establish routing tables based on the names, and direct requests to the closest available copy. This is a natural fit since routing mechanisms are expressly designed to find shortest paths while avoiding failures, and those are the keys to providing availability in name resolution. This approach essentially turns name resolution into name-based anycast routing.

However, such approaches faces severe scalability challenges. Large-scale experiments on GENI would help researchers understand the extent to which the approaches being contemplated could achieve the requisite scales under realistic loads, while also tolerating real-world failure scenarios.

One of the concerns with BGP is that it does not provide adequate levels of security. GENI can be used to evaluate the route-computation service of a **security-enhanced alternative to BGP**, for example, [HU04]. For example, upon learning a BGP route announcement from a neighboring domain, the service can classify the route as "suspicious" if the Autonomous System originating the route does not agree with past history. The service can prefer non-suspect routes over suspect routes. The experiment could evaluate this new variant of the BGP path-selection process and see if it effectively protects the experimental network from attackers injecting false routing information.

Today's routing protocols have some severe flaws with respect to management. A FIND proposal titled *A Framework for Manageability in Future Routing Systems,* by Guerin, Zhang, and Gao, looks at the interplay of network routing and **network management**. The Internet partly owes its success to choices it made implementing routing in a manner that is both scalable and resilient, and in doing so it heavily relied on distributed mechanisms. The choice of a distributed solution, however, comes with its own suite of problems when providing visibility into how decisions are made, and therefore enabling manageability and efficient trouble-shooting.

Motivated by the need to provide better visibility into the underlying decision processes, a number of more centralized solutions are being considered as possible replacements for parts or all of the current Internet routing system. However, while centralized solutions improve visibility, it is often at the cost of scalability and not without its own challenges, in particular, to ensure timely and reliable distribution of decisions across an entire network. Exploring and understanding the fundamental trade-off that exists between distributed and centralized solutions for routing systems and developing approaches that preserve their respective strengths while remedying some of their weaknesses is a key goal of this project. The output of this effort will be in the form of new algorithms and protocols for routing systems that strive to achieve the scalability of distributed solutions while offering a level of visibility, and therefore manageability, comparable to that of centralized solutions.

In addition to providing visibility for manageability and troubleshooting, another fundamental question is whether and how much information about the underlying network (e.g., failures) to expose to applications and services running on it; and how this revelation can be done in a scalable manner. Conversely, there is the question of how much control/input should applications and services have in the network's decision making processes (e.g., route selection) without affecting overall network stability and performance. Today's Internet bypasses this whole set of issues. It exports very little information to applications, and applications have minimal control over its decision making. As a result, quality-sensitive applications often resort to extensive probing to infer network conditions, and react to them, often belatedly. This not only complicates application development, but also imposes unnecessary load on the network. A critical question is, therefore, to determine if and how features built into future routing systems for manageability purposes can be leveraged to offer applications access to useful network information, for example, allowing them to issue queries on network status as well as be proactively notified of events of interest such as route changes or the location of failures.

Another example of the interplay of routing and management is that current routing schemes are not designed to take a component out of service gracefully. They can deal with a *failed* component, but there is always a transient glitch as the protocols compute new routes. Network operators today can deploy routers in primary-backup configurations, and avoid transient

glitches through a carefully composed series of manual management commands. These redundant configurations, however, are not cost effective at the edges of the network. If network operators know they are going to take a component out of service, it should be possible for the routing algorithm to plan for this so the users never see a momentary outage. In cases where backup redundancy is not deployed, this mechanism may require reconfiguration of the lower layer to virtually bypass the serviced component. A demonstration on GENI could evaluate the effects on end-to-end performance of a **graceful maintenance architecture,** which would make step-by-step coordinated changes in the configuration of network equipment at multiple protocol stack levels to prepare for removing equipment from the network.

3.2.4 Routing and congestion control algorithms

A number of variants of routing and congestion control algorithms, leveraging control theory, advances in game theory, and so-called "mechanism design", have been proposed in recent years. Some theoretical models appear promising, if analytical studies are correct, but these approaches must be tested in realistic settings. One reason for large-scale testing is that some algorithms assume correct information about routing topology or other network structure that is probably not reliably available in practice.

Other simplifying assumptions may also yield unrealistic or unusable ideas, however attractive the theory may be. Experiments of this nature would allow us to understand fundamental principles of optimization, control, and economic incentives, in a setting with inexact information. Improved routing and congestion control will have tremendous practical impact on world-wide network infrastructure.

While research over the past 10 years has addressed the basic interaction of end-to-end congestion control algorithms like TCP with router packet dropping schemes like RED, several new directions of research require new models and analysis methods. For example:

- What happens when buffer sizes are really small (and the previous theory doesn't quite apply because it makes a continuous approximation of an essentially discrete system and the fidelity of the approximation is poor in the "small-buffer limit")?
- What happens at the flow level? Current theory focuses on packet level models; we need to understand the behavior at the level of flows (flow completion times, etc) based on the underlying packet-level model.
- What happens in networks where packets cannot be dropped? Such networks are proliferating in Data Centers (e.g. Fibre Channel, Infiniband, Data Center Ethernet) where link-level pausing mechanisms enable switches not to drop packets. What effect does such link-level pausing have on end-to-end congestion control, especially when TCP relies on packet drops to regulate its sending rate?
- Finally, how good are the models at capturing the behavior of really large networks? I.e. how tractable and how meaningful; in short, how scalable is the theory?

Coming up with usable, large-scale theoretical models is challenging, requiring a combination of analysis, simulation and emulation. The implications of network operating policies which are based on "best industry practice," or "service level agreements," or "economic and security considerations" need to be easy to incorporate into the theory.

3.2.5 Security

As discussed in Section 2.1.1, perhaps the single most important motivation for rethinking the Internet is to improve its security and reliability. At lower layers of the protocol stack, the current Internet is plagued by undesirable traffic, including spam, DoS attacks, and malicious

Living Without Congestion Control

One of the most crucial aspects of the current Internet architecture is its reliance on individual flows to control their rates in order to avoid overall network congestion and achieve a reasonably fair allocation of bandwidth. TCP, in which flows slow down in response to packet drops, is the dominant form of congestion control today. However, there is a wide range of congestion control proposals in the literature that improve on TCP's performance, and extend its range to higher speeds and lossier environments. These proposals vary in the manner of adjustment, the type of congestion signal, and the nature of fairness sought, but they all share the notion that, to achieve low-loss and reasonably fair bandwidth allocation, flows should restrain their rate when the network signals them to do so.

Recently researchers have begun exploring a future without congestion control, in which flows do not attempt to relieve the network of congestion but rather send as fast as they can whenever they have something to send. If all flows are sending at maximal rates, then the packet loss rate within the network is quite high. To cope with this, flows use efficient erasure coding, so the effective bandwidth achieved by the flow is a function of the throughput rate of the flow, and does not depend on its drop rate. That is, almost all delivered packets will be useful, irrespective of packet loss.

This approach has several advantages:

- Efficiency: When end hosts send packets as fast as possible, all available network resources between source and destination are utilized. Furthermore, because links are constantly overdriven, any additional capacity is immediately consumed.
- Simplicity: Because packet drops (and reordering) are inconsequential, routers can be considerably simplified. For instance, routers no longer need to buffer packets to avoid packet loss, dispensing with the need for expensive and power-hungry line-card memory.
- Stability: this approach transforms the sender's main task from adjusting its transmission rate to selecting an appropriate encoding. Unlike the former, however, the latter has no impact on other flows. Hence, in this approach, traffic demands are fundamentally more stable than with traditional congestion control algorithms where the frequent rate variations may influence the behavior of other flows sharing a bottleneck.
- Robustness: Using this approach, end points are forced to cope with high levels of loss and reordering in steady state. As a result, the network can drop, reorder, or duplicate packets without severely impacting flow throughput. Also, due to the flow isolation described above, the end points of one flow need not consider the congestion due to others when transmitting, so greedy parties cannot manipulate the network against each another.

Perhaps the most intriguing possibility raised by this design is the chance to employ bufferless all-optical switches. Early results indicate that network efficiency can be maintained with extremely small or even non-existent buffers in switches, removing one of the chief barriers to the deployment of all-optical cross-connects. However, in order to provide fairness, the switches would need to drop packets selectively, akin how it is done in, say, Approximate Fair Dropping (AFD). This is an area that has not been well explored in the optical switch literature, but would be essential for making this approach viable.

GENI would provide an ideal platform for experiments with this approach. It would allow novel optical-switch technology to be employed in conjunction with new host behaviors, and would also prevent the highly congested behavior engendered by this experiment from affecting other users. In

addition, it would allow this approach to be tested in a variety of contexts (e.g., wireless subnets) where its performance may be more problematic.

traffic routed through compromised (zombie) hosts. At higher layers, applications are plagued by various attacks that might be mitigated by security mechanisms at the application layer and lower layers. To date, stopgap measures to fight undesirable traffic via add-on security mechanisms have not been successful. This is not surprising, as the problems stem from two fundamental shortcomings in the design of the Internet: there is no way to reason about the properties of hosts on the edge of the network, thereby assuring the routers of the validity of the traffic emanating from a given host, and there is no way to reason about the properties of services provided by the network, thereby assuring the edge nodes of the integrity of the network fabric. The overarching goal of a new Internet is not just a collection of security mechanisms but an overall architecture for security, which is woven into an overall design for a network. This will require development and experimental evaluation of individual mechanisms, user services, and combinations of components, on a large scale.

Since security means resilience to malicious attack, security studies involve identifying the essential properties that must be preserved in the face of attack, and the *threat model,* which includes the set of actions that an attacker might use to degrade these properties. While it is difficult to determine experimentally that a system is secure against all possible attacks within a specific threat model, a number of important security questions can and must be addressed experimentally before security improvements can be accepted and adopted in widely used networks.

The characteristics of innovative networks and innovative networked applications that can be evaluated experimentally include:

- *Performance*. How well does the system perform under different loads? What are the performance penalties associated with different levels of security, achieved in different ways?
- *Usability*. Is the secure system, providing a specific level of security and functionality, attractive to users?
- *Resilience to known attacks*. How does the experimental system respond to known attacks, carried out in realistic ways?
- *Resilience to new attacks*. Can creative attackers, such as researchers from the security research community, interfere with the operation of an experimental system, while it attempts to serve a developing user community?

The benefit of focusing on security when redesigning the Internet, rather than just adding it on to an existing design, is that features built into the architecture can dramatically alter the range of what can and cannot be achieved in terms of security. For example, the theory/cryptography community has proven that for many protocol problems, achieving security in a concurrent environment like the Internet *requires* some sort of common infrastructure, such as shared randomness or a weak form of a PKI [BAR04, CAN02, YEH04]. Other types of architectural primitives, such as anonymous channels, quantum channels, or high-rate sources of

randomness, have been shown to allow for cryptographic protocols whose security is unconditional (i.e. does not rely on the assumed intractability of problems like integer factorization), cf. [AUM02, DAM99, ISH06]. As the cryptography community continues to explore the theoretical possibilities raised by these and other security-relevant architectural features (such as secure logging, micropayment infrastructure, and source authentication), GENI will provide an exciting opportunity to also compare these features in terms of efficiency, cost, and compatibility with other design considerations for the new Internet.

Limiting Collateral Damage

As long as we cannot write perfectly reliable software, completely free from vulnerabilities, the threat of compromises to end hosts, servers, and network routers will be an ever-present danger. Improving software reliability will, of course, continue to be a major research focus, but the inevitability of compromised nodes in the Internet is a serious problem that any future architecture must address. Thus, in addition to reliable software, a key goal for any future Internet will be *containment*; i.e., the ability to limit the collateral damage caused by such compromises. The current architecture is especially fragile in this regard: a single compromised router, whether by malice or accident, can divert or "black-hole" a significant fraction of Internet traffic; collections of compromised hosts can be (and are) used for nefarious purposes such as distributed denial-of-service (DDoS) attacks, spam, and email phishing.

A concrete goal would be to develop a set of architectural principles and implement a design that can demonstrably improve the Internet's ability to limit collateral damage. Because one of the problems in the Internet today is a lack of identity, it is easy to spoof addresses and hijack routes; the latter directly causes damage, and the former enables the source of danger to be hidden. Thus, *accountability* – being able to identify a responsible component in the system for any action, as well as the ability for that component to demonstrate deniability for any ill action it did not actually commit – can provide a much firmer foundation for such identification, and should be a part of the future Internet architecture.

One approach could be to make all network addresses and names *cryptographically verifiable*, deriving them from their public keys (using hash functions). This notion of self-certification, developed earlier in the context of HIP and SFS, would explicitly name both the administrative origin and the unique end-point identifier for the host. For example, addresses could be of the form AID:EID where AID is the identifier of the host's AS and EID is the host identifier, with AID and EID being hashes of, respectively, the AS's and host's public key. Such an addressing structure would make any spoofing or forgery of source addresses detectable without relying on cumbersome and error-prone manual configuration of egress filters. It would also make route hijacking and other security compromises to inter-domain routing impossible, and would do so without requiring PKIs or third-party certification.

In addition, once such firm notions of identification are in place, one can leverage the good intentions of most host owners and operators. (This is a safe assumption since most attacks are launched from compromised machines whose owners are well-intentioned.) Network interface cards can be modified to provide some low-level functionality that is not under control of the host operating system (but could be initially configured by plugging the NIC into a USB or serial port on the host). Such intelligent NICs would be beyond compromise by external intruders, but could limit the extent to which that machine participated in attacks on other sites. For instance, if host A did not want to receive any more packets from host B, host A could send a packet to host B's NIC requesting a time-limited cessation of packets sent to A. The success of such a scheme relies on the cryptographically secure notion of identity (so that B knows that it is A making the request) and on the ability of the network to prevent spoofing (so that A knows the attacking packets are coming from B). Thus, providing accountability, by tying addresses tightly to notions of identity, would enable much stronger DDoS prevention measures.

It would be extremely difficult to verify such an approach in today's Internet, given that it calls for extensive changes to addressing, routing, and hosts. However, on GENI one could establish a network running this new architecture and provide hosts with a software proxy that would imitate the behavior of the intelligent NICs described above.

To test the robustness of these methods against attack or evasion, one could recruit attackers whose goal would be to overcome these defenses. These attackers could be officially organized "Red Teams" that are explicitly funded for this purpose (and who could be seeking to attack many of the security designs being tested on GENI). Or the attackers could be implicitly encouraged by the offering of prize money for any successful compromise of the experimental machines. In both cases, the viability of these experiments relies on GENI's ability to set up such an alternate architecture and keep it isolated from the rest of the ongoing experiments, so that any successful attacks on it do not threaten other experiments.

An important thread through many likely security-related experiments is the trade-off between usability and security. Experience with security mechanisms has shown that many ways of strengthening a system against malicious attack make the system less convenient to use. This trade-off can be expected in future systems, since many security mechanisms must distinguish between honest activity, of the sort the system is designed to support, and malicious activity that is intended to disrupt the system. Although no fundamental theoretical tradeoff has been proved, it generally becomes easier to distinguish honest and malicious activities if honest users take additional steps to distinguish themselves or their actions. Because of the often-observed trade-off, a key goal in security experiments is to evaluate the usability of a system, by representative individuals with no vested interest in the success of the system, in parallel with experiments aimed at determining the resistance of the system to malicious attack. Privacy is another important goal that requires experimental user communities on a substantial scale. Some forms of security, including any mechanism that makes decisions on the basis of trust, reputation, or authority, will require identity schemes, which must be carefully conceived to balance issues of privacy and freedom from excessive oversight with the goals of accountability.

There is a wide range of important and informative experiments that can be conducted on the GENI facility as currently conceived. Examples, some of which are described in more detail below, include:

- Spam-resistant email,
- Distributed decentralized access control,
- Worm propagation and mitigation,
- Reputation systems,
- Improved network infrastructure protocols,
- Selective traceability and privacy,
- SCADA simulation,
- Botnet and overlay network security and detectability,
- Economic incentives in network infrastructure and applications,
- Light-weight security tools and algorithms for low-power computing devices,
- Anonymity in routing and applications,
- Privacy-preserving data-mining,
- Secure multi-party communication,
- Proof-carrying code to protect hosts from malware (and other purposes),
- Secure electronic cash and micro-payment mechanisms,
- Experimental combinations of security mechanisms for improved enterprise security.

Spam-resistant email: Unwanted bulk email, or SPAM, is a pressing and widely recognized problem [CRA98, DWO03]. While mechanisms such as S/MIME [RAM99] and Sender Policy Framework (SPF) [LEN04] have been proposed, no effective defense has achieved widespread adoption. An experimental email infrastructure, perhaps compatible with existing client applications, could be set up in parallel with existing Internet email. Designers of one or more such systems could set up their email infrastructure and invite users. While interoperability between an experimental network and the Internet may still provide a point of attack for spammers, an experimental network might still provide reliable, authenticated email between users on the experimental network. As the experimental network grows, its value may become apparent, and solutions to spam may result. Clearly an experimental system of this sort must be left open to malicious attack, in order to test its robustness. Since early adopters of an

experimental email network may be too few in number to attract commercial spam, it may be desirable to offer rewards to successful infiltrators, for example. While it is not clear whether authentication, reputation mechanisms [GYO04], or other concepts will provide the best solution, spam is a sufficiently pressing problem that experimental alternatives to current email are worthwhile and attractive.

Worm propagation and mitigation: Worms have become progressively more sophisticated in recent years [MOR03]. The increasing speed of worm propagation means that future worm defenses will have to act more and more rapidly, to the extent that future worm defense mechanisms must be able to detect unknown worms rapidly [BER03, SIN03]. GENI provides an ideal setting for designing and testing worm detection and mitigation methods, allowing larger scale experiments, and open challenges to communities of malcode developers.

Selective traceability and privacy: One of the issues discussed in Section 2 is the tension between privacy, anonymity and freedom of action on the one hand, and accountability and deterrence on the other. One dimension of this is a social discourse on the desired balance. But there is also a strong technical component—the invention of techniques that may give us "more of both". Schemes that provide a predictable degree of privacy, but also allow some traceability under sufficient authorization, may be able to improve the range of social choices that a future network can provide. This sort of objective should be a challenge to the security community, including cryptology. The FIND proposal titled *Enabling Defense and Deterrence through Private Attribution,* by Snoeren, Savage and Vahdat, proposes a novel way to attach a strong source identifier to a packet in such a way that privacy is preserved under normal circumstances, but identity can be revealed when deemed necessary. These strong identifiers can be used both after the fact, and in real time to constrain the communication patterns of the network to willing groups of communicants. The scheme uses a packet attribution mechanism based on group signatures that allows any network element to verify that a packet was sent by a member of a given group. Importantly, however, actually attributing the packet to a particular member requires the participation of a set of trusted authorities, thereby ensuring the privacy of individual senders.

Assurance of uncompromised endpoints: One approach is to leverage trusted computing (as in TCG) that enables a qualitatively different type of assurance for computer systems. Attestation enables hosts to provide tamper-proof certificates of program properties. Employed at the core or at the edge, these certificates can be used to rule out much unwanted traffic. For example, spam messages and DoS attacks can be ruled out via "human presence" certificates attached to sessions or packets, zombies can be identified via secure program identifiers that reveal whether a program has known security holes without revealing or constraining which version of a program users are allowed to run, network services can be securely identified and selected, not via blind trust, but through intelligent reasoning about their attested semantic properties. A trusted system using attestation can provide a lower layer underneath GENI hosts to provide compartmentalization between different network services, architectures, and applications implemented within the network, as well as to provide a secure, tamper- and masquerade-proof identification of these properties.

Overall, trusted computing approaches enable remote parties to find out about the properties of a software stack, which is a key enabling technology for any intelligent trust decision. These are investigations into the foundations of distributed trust, exploration of cryptographic primitives,

and hierarchical trust statements. If successful, the results will be trustworthy network services and infrastructure. [AND03a, GAR03a, GAR03b, SES05, SHI05, TPA]

Access-controlled Routing: An example of an approach to security is Capability-based Routing Access Control. In the current Internet, a packet can be sent on the basis of knowing an IP address. An alternative is to only allow connectivity when some access control mechanism deems this appropriate. In a capability-based approach, for example, all connectivity is granted by handing out capabilities. A capability is an encrypted source route between any two communicating end points. Capabilities are constructed by a centralized Domain Controller (DC) that has a complete view of the network topology. By granting access using a global vantage point, the DC can implement policies in a topology-independent manner. For example, the DC is able to restrict or grant access to a certain file server no matter how the connection is routed. This is in contrast to today's networks where firewalls are implicitly tied to topology that can become complex as the network grows. Access control for a single system is a well-studied problem. However, scalable distributed access control for network connectivity is an unexplored area with significant challenges. Using this approach, administrators are free to implement a wide variety of policies that vary from strict to relaxed and differ among users and services. The key here is that connectivity control allows the easy implementation and enforcement of a rule once simply expressed. [AND03b, CAS06, GRE05, REX04, ROS03]. A FIND proposal titled *Designing Secure Networks from the Ground Up*, by McKeown, Boneh, Mazieres and Rosenblum allows routers to enforce security policy by requiring all traffic to explicitly signal its origin as well as intent to the network at the very outset. The solution is tailored for enterprise networks, which often enforce tight controls on who can communicate with whom over their networks. The proposed solution for private networks requires all network-wide policies to be specified at a single location called the domain controller. If the policy allows it, the domain controller grants explicit permission for users to communicate. Their proposal includes the development and demonstration of a router that can check security permissions at line speed.

Implications for GENI infrastructure: Since attacks against one experimental system must not interfere with other experiments, or the ability to monitor and evaluate the system, security raises some difficult challenges for the design and implementation of the GENI facility. The techniques that are used inside GENI to isolate different experiments, the techniques for data gathering and experimental monitoring, the techniques for resource allocation all need to be designed taking into account the nature of security experiments.

3.2.6 Network Management

It is hard to over-state the importance of better network management. The Internet has always been notorious for being less reliable than the phone network. As more and more critical services are being moved over to the Internet, the need for rock-solid network management techniques also becomes critical. Even for non-critical services, the cost of systems management dominates IT costs and must be reduced. Finally, networks deployed on the fly, such as for military and emergency services, often do not work at all except under very controlled circumstances. All of these areas may be improved through a better network management architecture.

Network Management in the Internet today is, to put it generously, ad hoc. Network operators often find themselves using many different independent management tools, and are forced to specify device configuration at a very low level, with long lists of detailed configuration

information that must be specified and validated by the human operator. SNMPLink[3] lists more than 1000 management applications, many of them vendor specific command line or

[3] http://www.snmplink.org, a network management information site.

Centralizing Network Management

Managing a large data network is immensely difficult, as evidenced by two interesting statistics. First, the cost of the people and systems that manage a network typically exceeds the cost of the underlying nodes and links. Second, more than half of network outages are caused by operator error, rather than equipment failures. Managing a data network is challenging because the existing protocols and mechanisms were not designed with management in mind, forcing network operators to rely on limited information about network conditions and indirect (and sometimes inadequate) control over the behavior of the network. Retrofitting network management on the existing architecture has proven quite difficult, and has led to a plethora of tools, scripts, and procedures that add an additional layer of complexity on top of an already complex network while providing a level of management that is far from ideal.

Given the importance of network management, it seems clear that any future Internet should be based on an architecture that is intrinsically easier to manage. To see what such an architecture might look like, it is useful to first revisit today's division of functionality between the data, control, and management planes. Today's data plane implements packet forwarding, filtering, buffering, and scheduling on routers; the control plane consists of the routing and signaling protocols the routers use to coordinate their actions; and the management plane collects measurement data and tunes the configuration of the routers to achieve network-wide objectives. The fact that the control plane doesn't explicitly include management functions essentially forces the management plane to "invert" the operations of the control plane in the search for meaningful changes to the network configuration. For example, to achieve the desired access control, operators need to know which path packets will take so they can be sure that the corresponding access control state is configured on the appropriate routers.

One approach to a management-oriented architecture would be to largely eliminate the control plane, and move nearly all functionality (besides forwarding packets and collecting measurement data) out of the routers into a separate management service. The resulting "4D" architecture [REX04] has four planes: a data plane (much like today), a discovery plane (that collects measurement data), a dissemination plane (for communicating with the routers), and a decision plane (to manage the network). The decision plane directly configures the data plane. In order to make these configuration decisions in a coherent and informed manner, the decision plane would be logically centralized and would maintain a network-wide view of the topology and traffic. In practice, the decision plane would be implemented as a collection of servers, to reduce vulnerability to failure and attack.

Any evaluation of the 4D architecture must consider two main questions:

- *Is the 4D architecture technically feasible?* Moving a network's decision logic to a small collection of servers raises natural questions about reliability, security, and performance. These are classic "systems" questions that cannot be answered by simulation alone, and must be evaluated by a prototype implementation operating under realistic conditions.
- *Does the 4D architecture reduce management complexity?* This question also necessitates a realistic evaluation of an operational system over a sizable period of time, and would involve having the prototype interact with other domains so that both the intra- and inter-domain aspects of the design could be fully tested.

Thus, these questions can only be answered using a deployed prototype, and GENI provides the necessary infrastructure for such experiments. The aspects of GENI that are essential for testing 4D are real user traffic, exposing management functionality of network resources, the ability to run long-running experiments to gain operational experience, and the ability to construct multiple interacting domains. In addition, evaluating the effectiveness of the centralized control requires realistic delays between the servers and routers, which may not be feasible in an overlay.

HTML-based tools. It is not uncommon for a network device to have thousands of manageable objects. MIBDepot[4] lists 6200 MIBs (Management Information Base) from 142 vendors for a total of nearly a million MIB objects. A single ISP backbone router configuration file can consist of more than 10,000 command lines. A recent IT industry survey claimed that 80% of the IT budget in enterprises is devoted to maintain just the status quo - in spite of this, configuration errors account for 62% of network downtime.[KER04]

Arguably, network management is the way it is because the Internet never had a cohesive architecture for network management in the same sense that it has an architecture for data-plane protocols. For instance, there is no management analog to the service abstraction that layered data-plane protocols provide for each other. One reason for this might be that the original Internet architects had their hands full simply with getting the data-plane working. Another reason might be that the early Internet was much simpler and smaller than it is today, and the users of the early Internet were themselves networking experts. Or perhaps there simply is no simple service abstraction for network management. Perhaps network management is architected about as well as it can be.

Broadly speaking, we define network management here to be the configuration and maintenance of networks. This encompasses network planning, configuration and provisioning, staging and testing, operational cutover, failure detection and correction, and troubleshooting and repair. These activities apply to establishing connectivity (getting packets from here to there), security management including filters and user authentication, and performance management. To bound the problem, we do not define per-flow activities as part of network management. In other words, TCP congestion control, setting up an IPSec session, or establishing an RSVP flow are not part of network management as defined here.

In the following paragraphs, we discuss a number of the basic design decisions required for a network management architecture. Though discussed separately, clearly these all relate to each other and therefore can't be considered in isolation.

> **The management channel**: One of the foundational principles in managing telephone networks is the use of a telephone wire that bypasses the switching equipment, allowing the crafts-person to communicate even when the network is broken. The IP Internet has never had such a management channel: SNMP[HAR02], for instance, requires that IP be up and running before it can operate, and is therefore useless for dealing with many low-level failures. Recently some researchers have proposed a separate management channel called 4D [GRE05] that uses the raw circuits of the Internet, but which operates below the IP layer and allows network managers to securely discover and communicate with network equipment. Such an approach, however, is likely to have its own failure modes and security holes. Operational experience with network management channels is needed to understand their pros and cons. A current FIND proposal, *Towards Complexity Oblivious Network Management*, by Francis and Lepreau, extend prior work in this area to deal with cross-domain issues.
>
> **Manual versus automatic configuration**: Internet researchers have always been attracted by self-configuring control algorithms, especially routing algorithms. The

[4] http://www.mibdepot.com, another network management information site.

ideal of a network that can configure itself and react to failures without human intervention is very appealing. Network managers, however, are rightfully wary of

Unified Traffic Engineering

ISPs devote much effort to traffic engineering, which is the task of distributing traffic among an ISP's various links in a way that efficiently uses their networking resources while still meeting their service level agreements (SLAs). Thus, traffic engineering is crucial to the effective and profitable operation of ISPs, and ISPs understandably devote significant energy and engineering to this task. Since the original Internet architecture did not address this issue, current traffic engineering techniques have evolved in an ad hoc manner, with traffic being controlled by three different entities: hosts, routers, and operators. Host-based congestion control adjusts the rate at which sources send in response to indications of congestion; routers direct traffic over shortest paths, as measured by the configured link weights; and operators set these link weights based on previous measurements of traffic matrices in an attempt to achieve their utilization and SLA goals. Unfortunately, the task of finding an appropriate set of link weights is computationally intractable, forcing operators to resort to heuristics. Moreover, these various control loops are not coordinated, in that end hosts adapt sending rates assuming routing is fixed, and operators tune link weights assuming that traffic is inelastic. In fact, recent research shows these control loops interact in a highly suboptimal manner.

A more attractive alternative would be to provide a coherent traffic engineering architecture in which a single set of mechanisms could allow both users and operators to meet their respective goals. One possible approach to achieving Unified Traffic Engineering (UTE) would be to take a top-down approach that starts with a shared objective for operators and users. A distributed architecture respecting the following separation of concerns can then be derived:

- Operators provide multiple paths for sources
- Links set "prices" based on local load information
- Sources adjust their sending rate over each of the multiple paths based on this price information.

This approach unifies the various control loops, and provides a way in which users and operators can both achieve their goals through direct manipulation of their configurations. From a theoretical perspective, optimization theory guarantees that a wide variety of adjustment algorithms will converge to the optimal solution at equilibrium.

Simulating this approach has helped identify some of the important design issues. However, there is a large gap between idealized simulations and practical deployment. The proposed architecture is difficult to deploy today since several essential functions are not available. First, today's routing protocols are designed for shortest path (intradomain) or best path (interdomain) with extremely limited multipath capabilities. Even where multipath routing is supported, the flows are split evenly amongst the paths, and flexible splitting ratios are infeasible. Finally, traffic policing is required to ensure sources are not sending too aggressively and this is also not routinely supported today. Without evaluation in realistic settings, ISPs would be unwilling to make such changes to the current architecture to support a UTE approach.

Several aspects of GENI make it an appropriate setting to test UTE designs. First, because traffic engineering requires the allocation of bandwidth over links, UTE mechanisms cannot be tested on an overlay where the variations of available point-to-point bandwidth would interfere with the dynamics of the traffic-engineering mechanisms. Thus, GENI's ability to dedicate fixed capacities to various experiments would be important to making these experiments useful. Second, GENI's ability to embrace real user traffic is crucial, as the success of any UTE scheme will be its ability to adapt to traffic variations in real time and evolving user requirements over longer periods of time. Lastly, GENI's topological scale and scope, which are approximately that of a reasonably-sized ISP, will

provide a necessary degree of realism.

such dynamic systems. When they fail to work properly, it can be extremely difficult to know what is going on and fix them. For example, experience with the ARPANET showed that it is very difficult for distributed routing algorithms to respond dynamically to traffic load without going into oscillations [KHA89]. Early experience with BGP taught us that extensive manual configuration is necessary to prevent bogus route advertisements. Having said that, it is clear that some dynamic responsiveness is necessary. When a router fails, packets must be rerouted as quickly as possible. There has lately been a trend towards localizing network dynamics, for instance fast rerouting within otherwise statically configured routing tables [PAN05]. Fundamental research is required to identify the correct balance between automatic and manual control, to allow automatic controls to be themselves controlled or constrained by manual controls, and to allow operators to have visibility into the operation of these automatic controls. Extensive testing and experimentation will be required to develop confidence that these proposals are useful.

Centralized versus distributed: There are trade-offs between centralized versus distributed management. Centralization tends to be easier, while distributed is generally viewed as more robust and scalable. Internet routing algorithms have historically been distributed, but recently researchers are suggesting that we should take another serious look at centralization [CAE05]. This change in thinking is driven by two factors. First, in practice there turns out to be far more static configuration required than initially envisioned, especially the policy information associated with BGP. Centralization simplifies this configuration. Second, computers have become more powerful over the years, thus making it feasible to centralize management decisions. At the same time, the simple fact that different networks are operated by different administrations means that complete centralization is obviously impossible. In addition, it may still not be feasible to centralize management in resource-constrained networks like ad hoc mobile radio networks. Given the experience gained in several decades of Internet operation, this is a good time to rethink the centralization versus distribution equation on a solid experimental basis.

Abstraction: One of the central tenets of computer science is that complexity can be reduced and managed with abstraction. Simple interfaces allow complex functions to be used by programs that don't understand the details of those functions. Remarkably, network management completely lacks a decent low-level abstraction. This is not to say that there aren't standardized management interfaces: SNMP is one such interface. SNMP, however, does nothing to abstract away the complexity of protocol operation. Every managed protocol object (counter, parameter, etc.) is exposed, forcing network management functions to cope with all the complexity of protocols and their interaction. There has been considerable attention paid to reducing the complexity seen by the human manager. For instance, a key selling point of products like HP Openview is that it allows non-experts to manage large networks. But these products must still cope with a complex network, and often fail to hide much of the network complexity, especially for cutting-edge network devices.

A fundamentally different approach to network management is to have each protocol abstract the details of its operation into a small set of basic objects and primitives common to all communications protocols [HIT06]. Protocols could present such an abstraction to external network managers as well as to other protocol modules – those above and below it in the protocol stack, and those in other network devices. The

fundamental research challenge, of course, is to identify what the useful and effective abstractions are. An abstraction of a failure, for example, might allow a lower layer protocol to tell a higher layer protocol that it has lost connectivity, and that it can repair itself within 100ms, thus allowing the higher layer protocol to hold off on its own, more costly failure response. If the lower layer protocol fails to repair itself, it can then notify the upper layer of this. It is not at all clear that a good protocol management abstraction exists: one that successfully hides most protocol complexity without significantly limiting control over that protocol.

A FIND proposal titled *Design for manageability in the Next Generation Internet,* by Barford, Banerjee and Estan, proposes to define **network management building blocks.** The objective of the work is to discover and define a higher-level abstraction to specify objectives of management, and to equip all the components of a future Internet with embedded capability for management. They will focus on building blocks for ubiquitous measurement, data sharing, end-host signaling, event detection and data organization and presentation.

Cross-domain management: It is often the case that network failures in one network domain produce symptoms of failure in another network domain. Debugging these cross-domain network problems has historically been difficult in the Internet [FEL04]. Some of the reasons for this may be purely social---network operators are quick to assign blame to other networks. Another reason, however, may be that today there exists no good way to balance privacy concerns against the need for cross-domain network management. For instance, a network operator would never give a competitor free reign to its SNMP MIBs. There is no easy way, however, to limit a competitor's view to only those SNMP objects that are likely related to a failure. This fact implies that cross-domain interaction may need to occur at a more abstract or higher layer, a layer that currently does not exist in management systems. In practice, today human network managers in different domain cooperate on an informal quid pro quo basis. Such an approach, however, slows down problem resolution since a manager in one network must get the attention of a manager in another network to make progress. Worse, a quid pro quo approach only makes sense between roughly symmetric networks. A network manager in an enterprise or home network cannot expect to have an informal relationship with a network manager in its provider ISP. GENI, with its multiple interacting domains, provides a unique opportunity to experiment with new cross-domain network management approaches.

A FIND proposal titled *Model-based diagnosis in the Knowledge Plane,* by Sollins, Lehr, and Wroclawski, focuses on the problem of cross-domain management, in particular fault diagnosis where the root cause of the fault may not be within the domain where the failure is detected. This proposal is part of a line of research that explores a new framework for management called the Knowledge Plane [CLA03b], which aims to develop a set of protocols and conventions by which management agents in different

network regions, as well as agents on end-nodes, can communicate to achieve cross-domain management goals such as fault diagnosis and automated network configuration.

More than any other aspect of computer networking, network management "in the large" lacks a rigorous architectural foundation. The problem is complex and vast, and touches on all aspects of networking. The broad network management problem does not lend itself to analysis or simulation --- it is very much a systems problem. Individual sub-problems may be spacing problem attacked through analysis or simulation, but there must be a structure whereby such results can be tested in a broader context. As such, the research community needs to establish a framework within which individual network management research can contribute to a broader understanding. GENI represents a rare opportunity to test new management ideas, because GENI itself must be managed. As a shared and virtualized resource, there must be some form of cross-domain management. Even in the context of GENI, there will be a strong temptation to come up with quick-and-dirty management solutions so that other research can move forward. Part of the GENI ethos must be that non-management experiments buy into an experimental management framework so that management techniques can be tested in the wild. This in turn means that a framework for testing management ideas and integrating them with network experiments be in place when GENI goes live.

Beyond this, the operation of GENI itself present new management challenges. Specifically, in order for GENI to succeed, it must offer real services to real users. At the same time, experimenters must be able to take risks and sometimes crash their networks. The GENI infrastructure must be able to steer user traffic through certain virtual experimental networks, detect when performance through those networks is sub-par, and subsequently steer traffic away from those failing experiments. This problem alone requires new thinking about performance monitoring and cross-domain management.

3.3 Architectural implications of new network technology

In the following sections we look at several technology drivers that imply architectural divergence from today's Internet. Development of these technologies will drive GENI facility requirements. Each of the following sections considers the implications of the technology, some proposed experiments that could be conducted on GENI and facility capabilities that would be required.

3.3.1 Wireless Networks

The future Internet will include ubiquitous wireless connectivity. Wireless adaptive mesh networks and embedded wireless sensor networks will proliferate at the edges of the Internet and will enable novel applications and drive architectural requirements. Accordingly, we see the facility being used to both:

- Develop novel applications and deploy them at scale to understand what services and systems components would be required in a future Internet, and
- Design, prototype, and evaluate novel architectural components and examine their performance, flexibility, and manageability

In this section, we give examples of technology and its implications, and experiments that arise as we deal with these issues.

3.3.1.1 Mesh Networks

We use the term "wireless mesh networks" to refer to networks built out of nodes with radios (e.g., 802.11) that communicate with each other to form an ad hoc "self-configuring" network without much manual involvement. These mesh networks can bring broadband-quality Internet access to users who are not well-served by wireline broadband ISPs. In addition, the protocols

Cognitive Network Layers for Disaster Scenarios

Advanced communications services in disaster recovery operations have become a crucial element in the overall response, as there is a desperate and immediate need for information so that first responders can communicate with victims and each other, and authorities can coordinate overall response efforts. An emergency network formed for such a situation must provide robust and flexible communication service under extreme conditions. The networking assets involved in forming such emergency networks will be owned by disparate entities, spread across different geographical locations and spanning multiple policy domains, thereby adding further complexity.

Cognitive radios offer the promise of quick establishment of communication infrastructure without detailed preplanning. The extension of cognitive radio capabilities to the network layer, providing multiple network services within a framework supporting mobility, and providing a security framework for accessing these services are key components in providing solutions to demanding networking environments such as disaster relief.

The network layer protocols for such cognitive radio networks are still in the nascent stages of development, and there is an urgent need to assess the ability of these new approaches to provide application robustness, performance, and security. Because simulation and emulation are not sufficient to explore the capabilities of these cognitive radios in disaster response and other multi-application mobile environments, realistic experiments with a variety of prototypes are needed.

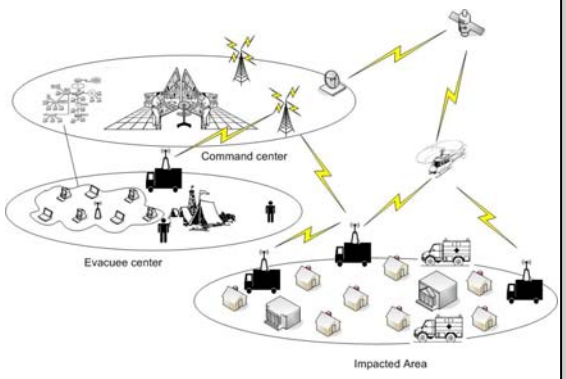

Figure 1. A disaster relief scenario.

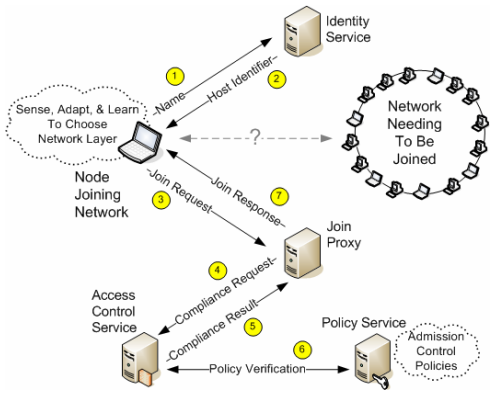

Figure 2. Node joining GENI cognitive network

Today's experiments in these areas are limited to local testbeds, which are often small in number of nodes and extent, and unrealistically homogeneous. GENI provides a unique capability to introduce heterogeneous environments separated by real world delays that realistically model disaster relief operations.

A testbed could be created using GENI with four different environments, illustrated in Fig. 1, specifically a Command Center to control and coordinate relief efforts (requiring one-one and one-many communications capabilities), the Impacted Area where connectivity is primarily provided by

heterogeneous wireless units deployed after the incident (supporting point to multipoint announcements, and point to point coordination with privacy), a rapidly growing Evacuee Center for those displaced by the emergency (needing broadcasts for finding relatives and point to point communications for requesting supplies, etc.), and the Relief Relay Center to catalog and direct relief (integrating mobile scanning and tracking devices to catalogue shipments of supplies, and provide directions to get supplies distributed). Each of these environments has its own service requirements, and all must be met within a coherent network design.

Fig. 2 depicts a possible emergency-services protocol that might be tested on the GENI facility.

developed on mesh networks for routing, channel access, error control, congestion control, and reliability are useful in wireless sensor networks.

There has been significant work in this area in the last decade, and technology developed in early research prototypes is now being deployed in practice. However, many research questions remain, only a few of which are covered here.

The most fundamental open question relates to capacity: how to design protocols that maximize the practically achievable capacity of these networks? This issue is well-understood in wired networks, but wireless channels have distinctive properties that make this question especially challenging. For example, over radio:

- Portions of a packet may be received correctly, but not the entire packet. Noise, interference, reflections, and obstructions affect the delivery of individual symbols (short bit-sequences) probabilistically.
- Concurrent transmissions by different senders interact at receivers in ways that are hard to predict.
- Each transmission is inherently broadcast and may reach or affect unintended receivers.
- Reception depends not only on transmit power and overall noise and interference levels, but also on the modulation and rate being used; since both power and modulation are controllable, the number of possible parameter combinations is very large.

It is becoming increasingly clear that achieving high wireless capacity requires a fundamental rethinking of traditional layering ideas. As one example, current physical layers demodulate received waveforms and provide a simple bit interface to higher layers. These higher layers have no information about which specific bits are likely to be correct and which aren't. Such information, if it were available, would enable higher layers to make better error control and forwarding decisions (e.g., forwarding only the correct bits). This information, however, is available at the physical layer, which usually has some information (depending on the scheme) about the "confidence" in any given demodulation decision. By propagating this information up as a hint, one might be able to achieve significant capacity gains.

This example is simply one among dozens of interesting ideas that might integrate – and then re-modularize – functions across the physical, link, MAC, and network layers of the protocol stack. Early work on opportunistic routing and multi-radio diversity has shown that taking advantage of probabilistic delivery and integrating MAC and routing functions can improve performance. Work on network coding has shown that for some workloads, combining packets across multiple flows can save on transmission bandwidth and improve capacity. Work on distributed spatial diversity and cooperative communication has shown that combining signal

information at spatially distributed receivers can improve capacity. In addition to these ideas, rate adaptation, adaptive modulation, and adaptive power control are all ideas that have been explored individually.

To date, these schemes have all been developed in isolation. It is very likely that the best practical schemes will combine these ideas in interesting and novel ways – this task is daunting, because of the number of degrees of freedom involved, but is also critical because of promised gains can be as high as an order of magnitude or more. GENI will enable researchers to both develop new schemes and compare against other proposals, a task that has proved near-impossible today.

Geographic Routing as an Internet Service

Using location information to optimize wireless networks has emerged as a powerful approach to scale capacity in high density or high mobility systems. In particular, geographic routing is a radically different approach to routing that provides far greater scalability than conventional ad hoc routing designs. Geographic routing does not forward packets based on Internet addresses; instead, destinations are identified by their geographic coordinates. The basic notion behind geographic routing is simple: routers forward packets to routers that are closer to the destination. However, there are many subtle complexities involved in avoiding "dead-ends" and overcoming radio anomalies.

Despite these complications, geographic routing holds great promise as it reduces the overhead of maintaining or acquiring network topology information and uses small routing tables. Geographic routing can be used to support vehicular applications such as content delivery or safety, as well as a broad range of location-aware mobile computing applications.

A basic research issue is that of evaluating the scalability of geographic routing in realistic large-scale deployments, and comparing the performance with more conventional overlay approaches. The location-aware protocols and underlying vehicle-to-vehicle MAC protocols supporting these applications have to date been primarily studied through simulation models. Only recently have small-to-medium sized testbeds such as the ORBIT outdoor field trial system become available, leading to new insights on the deficiencies of existing simulation models. For example, in an initial experiment with V2V georouting, researchers found that the underlying communication channel is significantly less reliable than assumed in simulations due to various radio effects. Thus, the V2V MAC and georouting protocols are currently under redesign to address issues of intermittent connectivity, deafness, and hidden node problems that arise under realistic conditions.

Current experiments on geographic routing are limited by the availability of large-scale testbed facilities. Key questions to be addressed to realize the vision of georouting for vehicle-to-vehicle and vehicle-to-roadside communication are reliability and latency of message delivery. These are determined by factors such as vehicle density, routes and driving patterns of vehicles, and message transmission rates. While initial results have been obtained with traffic simulations, moving this field forward requires large-scale experimental validation. The planned GENI facility would enable experimentation with such approaches at scale, especially through wireless edge networks with thousands of vehicles. Particular metrics of interest are latency, but also complexity and manageability of the resulting network structure. An implementation of the geographic routing stack and measurement instrumentation can be installed on programmable routers and access points in the planned city-wide urban grid deployment. Virtualization features in GENI would help to isolate this experiment from other services, and would thus allow measurement of end-to-end latency for geocast messages originating from both Internet hosts and moving vehicles. The setup would also support a study of georouter scaling properties in terms of routing table size, typical lookup delay and control messaging overheads.

With increasing stability of the prototype implementation, the network can also be made accessible

to pervasive application developers who can benefit from the geographic routing service. Feedback from this initial user population and system administrators who maintain the network will provide important insights on complexity and manageability of the system.

3.3.1.2 Cognitive Radios

Adaptive networks of cognitive radios represent an important and interesting research opportunity for both wireless and networking communities. Perhaps for the first time in the short history of networking, cognitive radios offer the potential for organic formation of infrastructure-less collaborative network clusters with dynamic adaptation at every layer of the protocol stack including physical, link and network layers. This capability has significant implications for the design of network algorithms and protocols at both local/access network and global internetworking levels. At the local wireless network level, an important technical challenge is that of defining a control protocol framework for cross-layer collaboration between radio nodes, and then using this control information to design stable adaptive networking algorithms that are not overly complex. At the global internetworking level, ad hoc clusters of cognitive radios represent a new category of access network that needs to be interfaced efficiently with the wired network infrastructure both in terms of control and data. End-to-end architectural issues of importance include naming and addressing consistent with the needs of self-organizing network clusters, as well as the definition of sufficiently aggregated control and management interfaces between cognitive radio networks and the global Internet.

Having an open-platform cognitive radio system in GENI will help the community explore a number of architectural issues towards understanding how this technology can be integrated into a future Internet architecture– these include control and management protocols, support for cooperative communication (this is sometimes called collaborative PHY, and network coding is one example), dynamic spectrum coordination, flexible MAC layer protocols, ad hoc group formation and cross-layer adaptation.

3.3.1.3 Intermittent and variable connectivity

While some radio links are highly stable and reliable, many radios today offer connectivity that is variable in quality and intermittent. As noted in section 3.1.4, this leads to the objective of a delay-tolerant architecture that can deal with these features. The wireless components of such a system require development of schemes for reliable delivery of large files over intermittent links, and push-pull architecture for mobile nodes, which enables opportunistic delivery of files, both to and from the wired network

The GENI facility could be used to explore a cache-and-forward architecture that exploits the decreasing cost and increasing capacity of storage devices to provide unified and efficient transport services to end hosts that may be wired or wireless; static, mobile, and/or intermittently disconnected; and either resource rich or poor.

3.3.1.4 Wireless communication among vehicles

Section 2.2.4 explored the implications of vehicular networks. There are many research questions raised by this vision, and an experimental vehicular network established as part of GENI can be used to explore many architectural and research directions, such as: radio and MAC layer performance assessment (e.g., download/upload capacity at Infostations at various speeds; car to car achievable data transfers [HUL06]); efficient use of the multiple 802.11p channels (control and data; prioritization of channels and data, etc); coexistence of critical and infotainment traffic; network protocol design and testing, including several new network protocols (e.g., epidemic dissemination, scoped broadcast, redundant forwarding control, multi-hop routing, network coding, congestion control, etc.); and, interfacing with the Internet infrastructure (coexistence of car to car channel with Mesh, WiMAX, 3G, 4G channels, smooth handoff across the available options, and interworking with the infrastructure to obtain support in mobility management, routing, traffic control, congestion control).

3.3.2 Optical Network Technology

Optical technologies will play a dual role in GENI, and it is important to keep these two roles conceptually separate. First, current state-of-the-art commercial optical technology will be an important component of the basic GENI infrastructure, providing a rich, malleable, virtualizable and high-performance backbone network. This will be explained at greater length in the GENI Facility Design document [DESIGN].

Second, the science of photonics is progressing at a rapid pace, and these new developments promise several exciting new capabilities. It is the prospect of these optical advances that is the focus here.

A short and necessarily incomplete list of the new optical approaches being pursued by researchers includes:

- Photonic integration to lower cost, power and footprint: Photonic integrated circuits (PICs) are densely integrated photonic chips with lasers, modulators, detectors and waveguiding regions.
- Integration between CMOS electronics and photonics to make manufacturable, low cost, lower power photonic modules: Technologies being explored today include integration of waveguides on silicon with optically active regions attached through wafer fusion or optical silicon bench technology.
- Novel all-optical switching technologies to enable scalable backbone virtualization, slicing, and dynamic reconfiguration: Two technologies available today are MEMs and Silica PLCs to enable higher degree ROADMs. Future technologies include silicon and InP photonic integrated circuits that put the complete switch and ROADM functions on single chip, driving down the power and footprint of this function by orders of magnitude over today's approaches.
- Optical Signal Management Technologies: Optical amplifiers (SOAs) as gain blocks and wavelength blocking (VOAs) to allow tunable losses.
- Tunable lasers to enable dynamic access to wavelengths on the network and lasers with decreased linewidth and phase noise: this will allow more advanced modulation and coding of the optical channel.
- Digital Optical Cascading Technologies to allow signals to propagate through more all-optical nodes with minimal network physical layer engineering: this enables all optical

3R (reshaping, reamplification, retiming) regeneration using mode locked lasers or photocurrent driven wavelength converters.

- Optical buffering and synchronizers to build networks of multiple nodes: Silica delay lines, wavelength dependent buffering and other techniques can be integrated on chip to build networks of many optical nodes.
- Coherent systems that maintain both the amplitude and phase information to enable more sophisticated modulation and coding techniques.
- Electronically controlled re-configurability at the chip level: Field Programmable PIC can be controlled by electronics FPGAs.
- New multilevel coding techniques: DPSK, QPSK enable modulation coding with more than one symbol per bit. These technologies will allow the GENI infrastructure to remain 10Gbps transport, with new technologies embedded in linecards that upgrade capacity to 40, 100 and 160 Gbps.

Dynamic Optically Circuit Switched Backbone Networks

Backbone networks are made from two parts: A physical optical transport network built from wavelength switches, TDM (SONET) switches, and optical patch panels; and an IP data network that uses the transport network to interconnect big routing centers. Large backbone routing centers are built from many small access routers (to multiplex the traffic arriving from customers at the edge), and a couple of big routers to direct the traffic over the backbone to three or four neighbors. The long-haul links are optical, and pass through many optical switching centers between the routing centers.

The transport network and IP networks are invariably owned and controlled by different organizations, either different companies (where one leases capacity from the other), or different organizations within the same company (where the transport network grew out of a very profitable telephony service). The transport layer is pseudo-static, with changes in topology taking weeks or months; the IP network is highly dynamic, with traffic matrices hard to measure and constantly changing. To accommodate uncertainty (from failures and changes in demand), IP networks are typically over-provisioned by a factor of four or more.

It has long been suggested that IP networks would benefit from directly controlling the transport layer so that they could provision new capacity very quickly on demand, perhaps within milliseconds. In one possible approach, the small access routers are connected together by a dynamic optical circuit switched (DOCS) network. A router copes with increased traffic for another access router by merely establishing a new circuit between them (or increasing an existing one). This allows the IP-layer network capacity to quickly adapt to changing demand. This approach could: have a profound effect on IP-level traffic engineering (perhaps eliminating it); change the routing protocols which no longer have a relatively fixed set of links to traverse; and allow the big routers to be replaced by fast optical circuit switches. It has been noted many times that, today, commercial optical circuit switches have about ten times the capacity-density (measured in Gb/s per cubic meter) of commercial routers, consuming about one tenth of the power, and costing less than one tenth per Gb/s. If this approach were deployed, it would be the first time we would reap the promised rewards of low-cost optical circuit switches (these could be, for example, fast MEMS-based patch panels, TDM switches or wavelength switches), and their almost limitless capacity with almost no consumed power.

If the opportunity is so enormous - and inevitable - why hasn't it happened already? To some extent it is already happening. The GMPLS standard (an IETF standard; there are equivalents from OIF and ITU) lays out a way for circuits to be established over a transport network, and some commercial circuit switches can be configured within seconds. DARPA has solicited the delivery of an ultra-fast provisioned optical network (CORONET), and some commercial networks now offer capacity on demand. The problem is that - in the absence of an elegant and simple usage model - the protocols and standards are bloated and complex. Without the means to test how users and operators would deploy such a network, no one is quite sure what needs to be built. Further, if there had been a means to experiment with dynamic optical circuit switching, it is likely that this approach would have been deployed a decade ago, when optical circuit switching was first possible and large routers were starting to be limited by power consumption.

Before such a design can be widely deployed, a number of questions need to be answered. First, how should an access router decide that it needs new capacity to connect it to another router; should it wait until the traffic changes, or should it try to predict it? How much capacity should it request, and what happens if the capacity is not available, is too expensive, or is needed for another circuit? What route should the circuit follow, and who should decide? These are just some of the many unanswered questions; and to answer them well will require an extensive series of experiments.

The GENI platform will support these experiments. The GENI backbone node architecture calls for a reconfigurable optical layer, in which TDM circuits, WDM circuits and whole fibers can be switched under the control of an experiment running in a slice. A user can deploy a conventional access router in the programmable router, which can aggregate traffic from tail circuits. When more capacity is needed, the router can signal the need for more capacity, and the optical layer can provide a new circuit across the GENI backbone network. Realistic experiments can thus be carried out quite easily, accurately emulating the way in which an optical circuit switched backbone network would be built and used.

All these approaches are being analyzed theoretically, and many of them have reached the stage of laboratory prototypes. For those approaches that produce exactly the same feature set as current technologies, but more cheaply and requiring less power, laboratory testing is mostly sufficient. However, the vast majority of optical approaches being pursued offer more advanced capabilities (such as the ability to rapidly establish new links), sometimes at the expense of other features (such as radically smaller buffers). In order to play an important role in any future Internet, such developments require an architectural response to take advantage of their new capabilities while overcoming any concomitant limitations. To pursue these issues within GENI, it will be essential to have these novel and experimental optical technologies accessible to researchers.

3.4 Distributed Applications

The experiments described to this point are associated with the design of the network itself, whether at the basic data transport layer or at a higher layer such as information dissemination. However, the range of experiments that can be carried out over GENI is much broader than this—it also includes advanced highly-distributed applications, and distributed application support tools.

3.4.1 Distributed Data Stream Analysis

Many Internet-based applications generate enormous volumes of data, including both the messages intrinsic to the application's operation, as well as "metadata" of various kinds generated in logs at the application's various sites. In many scenarios it is proving increasingly useful to monitor these distributed streams of data in near-real time, rather than wait for them to arrive in a "data warehouse" for post-mortem processing. Indeed, in many scenarios, it is simply infeasible to backhaul all the data to a centralized site, and in the absence of intelligent distributed analysis techniques, valuable information is discarded. To address this problem, distributed data stream analysis applications are being pursued in a host of scenarios including software system management, finance applications, real-time business applications (e.g., supply chain and fleet management), and distributed sensing applications for military, manufacturing and environmental settings. Technologies to do near-real-time data analysis have also been proposed to be used for monitoring tasks in core Internet management[HEL05] as alluded to in Section 3.2.6.

These applications often wish to provide distributed, communication-efficient, near-real-time analogues to the functionality currently available in centralized databases: query processing,

data-driven event triggering, and statistical data mining. Doing this at Internet scale requires fundamentally different technology than is available in database systems today because of (a) the need for *continuous* versions of these tasks that provide running results from streams of data, (b) massive distribution and attendant communication constraints, and (c) high aggregate data volumes, which preclude techniques that centralize and buffer entire data sets and make multiple passes over them. This raises a host of major intellectual challenges, including:

- *Distributed stream query engine architectures* that can run at Internet scales, with enormous aggregate volumes of data being generated across thousands or millions of sites.
- *Adaptive distributed query optimization techniques* that can map high-level, declarative requests into distributed algorithms, and continuously adjust the behavior and choice of algorithms as the characteristics of the data and the runtime environment inevitably change.
- *Approximation techniques* for queries, triggers and mining techniques that trade a small degree of answer accuracy for large savings in communication, typically by using "synopsis" or "sketching" techniques to compress data sets down to their key statistical properties [MUT06].
- *Secure multiparty data analysis algorithms* that allow queries, triggers, and mining tasks to be efficiently conducted by multiple parties across networks while both preserving data privacy and ensuring the veracity of results.

There have been initial efforts on all these fronts, but none of them have been explored deeply enough to achieve usable systems of any scale.

One exciting proposal is to explore these ideas in the context of a new Internet architecture, by collective monitoring of the network from its constituent end-hosts. More detailed challenge problems could be distilled by taking specific tasks used in modern centralized monitoring workloads (e.g. from ISP network monitors, or from enterprise intrusion detection systems) and attempting to scale them across a large prototype network under different models of distribution: e.g. a purely peer-to-peer approach involving only end-hosts, a hierarchy of carefully-situated monitoring infrastructure nodes, etc. The challenges could encompass not only a study of architectural and algorithmic efficiency, but also concerns about multiparty economic issues involving information flows, including incentives and mechanisms for providing misinformation or for participating inaccurately in distributed data analysis tasks.

3.4.2 High-Throughput Computing in Data Centers

In recent months, Internet services have announced the impending construction of "data centers" of unprecedented scale. Because these services for search, email hosting, maps and other features have become so prevalent, they form an intrinsic part of the Internet both qualitatively in terms of users' perceptions, and quantitatively in terms of traffic volumes.

The applications that run at these services are built upon massively parallel data analysis tasks. Data-intensive processing is often "embarrassingly parallel" (i.e. it can feasibly achieve linear speedup and scaleup), and hence it often pushes the frontier of high-performance architectures long before more complex algorithms (e.g. scientific simulations). Database research starting in the 1980's[DEW92] comfortably scaled tasks to dozens and even hundreds of machines on local-area networks using software building blocks like the Exchange operator[GRA90]; this work was widely commercialized in the database industry even while parallel computing companies

targeted at scientific applications went out of business. In the 1990's, Internet service research successfully harnessed and extended these ideas to run tasks at the scale of hundreds to thousands of machines[FOX97]; that work was widely commercialized by popular Internet services, using software building blocks like Map-Reduce[DEA04].

Will the next decade see hundreds of thousands or even millions of machines working together to do high-throughput data-intensive computing? It seems plausible, given appetites for increased data capture and analysis. However, basic questions at many levels of computing will need to be answered before achieving the next level of scaling.

For example, how will architects of both hardware and software navigate the boundaries between inter-processor networks (given many computational "cores" on a single chip), inter-computer "cluster" networks, and "The Internet" as we understand it today, and as it develops moving forward? How will these boundaries affect designers of high-performance protocols and systems, trying to maximize the throughput of data-intensive tasks across enormous numbers of computational components? As the computing platforms scale up, the software building blocks will have to adjust as well. For example, one distinction between the Exchange operator of the 1980's and the Map-Reduce tools 15 years later was the inclusion of a simple fault-tolerance mechanism in the latter – acknowledging the likelihood of component failure at larger scales. To achieve the next level of scaling even within a managed "data center", techniques from distributed computing and wide-area networking will have to integrate neatly with high-performance data parallelism, as the realities of partial failure and even adversarial participants play an increasing role even in data center applications. Many questions arise when mapping these techniques into high-performance data-parallel applications, particularly for tasks that stretch the limit of available computing power.

A new prototype Internet architecture will need to model data centers as a key component in the architecture. It would be extremely beneficial to develop a prototyping infrastructure to allow for the empirical analysis of a variety of alternative data center architectures, from as fine a grain as the "many-core" chip level up through the cluster level, to as coarse a grain as the federation of geographically-distributed data centers connected by long-haul links. Simple experimental benchmarks like the Sort benchmarks[NYB95] can help characterize the raw throughput and system balance of different parallel data analysis architectures; more complex applications like indexing, query processing and data mining can be tested on multiple architectural variants as a more robust empirical study.

3.4.3 Semantic Data Integration

Nearly all data-centric distributed applications have to deal with the challenge of semantic heterogeneity, in which concepts are described differently across multiple participating databases and software agents. This problem arises nearly everywhere, from inter-agency intelligence efforts in the federal government, to expenses in corporate mergers and acquisitions, to web information extraction, to the merging of simple address books across multiple desktop applications. Currently, post-hoc data integration typically requires significant, expensive manual work.

Traditionally, this problem has been tackled in one of two ways. Schema design and knowledge representation approaches have tried to provide tools, metaphors and disciplines to help designers develop rich information formats *before* trying to capture any data, in hopes of "getting it right" and duly flexible in advance. More recently, a great deal of research and

development effort has focused on the post-hoc integration of existing data in different representations.

This class of problem is almost certain to arise in the context of a massive new network design, both at the protocol level and at the application level. The original Internet carefully and slowly developed standard representations and semantics for mundane issues like packet headers, and for subtle but pervasive concepts like dates and times -- typically via a combination of both de facto standards from popular implementations, and via agreements from standards bodies. A next-generation Internet design may not have the luxury to develop common data representations slowly, by committee. On the other hand, leaving the design of common information to be frozen by the implementers of prototypes seems certain to lead to chaos.

As an example experiment, it may be useful to focus on relatively heterogeneous and data-rich services like **clusters of sensors** (i.e., regions of cooperating, networked sensors) that need to advertise service metadata. What is the service description language for such clusters? Does it describe the raw data they sense, or a more abstract query interface they expose? What control interface is exposed, e.g., over the rate of sampling in time, or the utilization of sensor power? Can a single schema or protocol cover very heterogeneous sensor patches, e.g. a cluster of environmental sensors in a forest on the one hand, and a set of automotive sensors in a fleet of vehicles on the other hand?

One answer to these questions is that such tasks are too application-specific to be handled by the network infrastructure. But is it wise for the network to abdicate any role in providing data or service descriptions? And if not, what role is feasible to provide in an extensible and fairly general way? These important questions should be at the center of any discussion of a newly redesigned Internet.

3.4.4 Architecture for location-aware computing

Location (defined in terms of geographic coordinates) is being recognized as an increasingly important aspect that needs to be integrated into mobile and sensor network applications. For example, mobile users seek geographically relevant information about people, products and services in the immediate vicinity[CSTB03]. Sensor applications require addressing of sensors by location rather than by network or MAC address. Vehicular safety applications require multicasting to nearby cars within a certain bounded region. In all these instances, techniques for naming, addressing and routing in the network need to be extended to account for geographic location. Techniques such as location service overlays and geographic routing have been proposed but never validated at sufficient scale or realism.

Experiments involving location will occur at various layers. A location-aware network experiment to be run on GENI involves instrumenting one or more wireless subnetwork with location determination services based on signal triangulation or other methods, along with implementations of overall or new network layer protocols for location service, georouting, etc. This experiment would start with a bottom-up validation of the accuracy with which location can be tracked by the network along with an estimate of protocol overheads and latencies associated with providing location information to higher layer protocols or applications. Once the protocol is validated and performance/overhead measured, it is anticipated that GENI would be used to offer long-running location services to new mobile and sensor applications with real end-users, leading to identification of one or more viable protocol designs.

At a higher level, we must design and validate a representation and semantics for storage,

propagation, protection and correlation of location information. An obvious representation for location is in terms of latitude, longitude and altitude, but for a first responder to a medical emergency, this has to be translated into address, floor, and room. There has been a lot of prior research that looks at geo-location, geo-tagging and so on, and we must determine which aspect of this should be a part of the network layer, which parts should be a common application support service, and which parts should be unique to each application. Answers to questions such as this will define a successful architecture for location-aware computing.

Location management is an excellent example of a design problem that will benefit from a multi-discipline approach, since an architecture for location management must take into account issues of privacy, ownership of personal location information, and rights of third parties and the state to gain access to this information.

One use for geo-location information is to drive a new sort of routing at the network layer. With the pervasive availability of location information, it is natural consider if a future Internet should integrate location information into the network architecture. One experiment would be to integrate a multi-resolution distributed location service, combined with trajectory-based forwarding as a key routing primitive. The location service builds a hierarchy of servers on the location registries available in wireless networks to keep track of associated nodes. Each node is associated with a home area, so that the location-service only needs to track nodes away from home. In addition, each level stores position information at progressively lower resolution, which improves both scalability (less updates) and privacy (less sensitive information). The trajectory-based forwarding mechanisms also allows for efficient coordinate system translations at routers.

One FIND proposal is looking at issues of geo-location at several of these layers: *A Geometric Stack for Location-Aware Networking,* by Geuteser and Martin.

3.5 Models and the theory of networking

GENI will facilitate an extremely general body of network experimentation. Theorists are excited about this prospect, and are interested in providing a formal, theoretical basis for what can and cannot be done with GENI. The modeling theory community poses two central questions: "Can GENI simulate an arbitrary network?" and in a similar vein: "What would it mean to provide a universal network?" where universality is in the Turing sense. At a more concrete level, theorists are starting from a basic, and essential, component of universality: being able to efficiently simulate an arbitrary network with entirely different naming and routing conventions.

Stepping up to a higher level of abstraction, a researcher may wish to embed a complex application or experiment into a target infrastructure. With GENI, will it be possible to take a multi-commodity flow problem with known routes and where the traffic matrix is known in advance and to provide an embedding that minimizes an appropriate combination of the resources the embedding consumes and the extent to which other experiments are interfered with? Answers to this question would necessitate practitioners working on systems issues of virtualization, emulation and repeatability to closely interact with theorists focusing on combinatorial optimization, graph theory and the design of efficient algorithms.

3.6 Putting it all together—architecture

The list of issues above, and the examples of approaches to deal with them, are only a very partial catalog of what the research community is preparing to do using the GENI test facility. It is important, as we consider this list, to remember to look at the whole and not merely the parts. Each one of the ideas above, and the many others that have been suggested by the research community, may be interesting in its own right, but the real payoff occurs when they are put together, their interactions explored, their joint implications worked out. It is through the combination and harmonization of many ideas like these that new architecture emerges. GENI can be used to support initial experiments to explore individual ideas, but the most important experiments on GENI will support the testing of these new architectures—combinations of these new ideas that greatly improve the fitness for purpose of the Internet.

4 The nature of experimental systems research

Section 3 has cataloged the anticipated range of experiments that can lead us to a future network. This section discusses the nature of the design and evaluation process that underlies the development of large, complex computer and communications systems, and explains why an experimental platform such as GENI is critical to validating these ideas.

As we discussed above, a future Internet will not be defined by one or two new features, but by the integration of a number of mechanisms into a cohesive whole, sometimes called an *architecture*. A future Internet will have to be evaluated relative to a large number of requirements, of the sort summarized in section 2. Looking at those requirements, we can see that the design and evaluation problem for a system of this complexity is marked by three characteristics that make it extremely challenging when compared to evaluation of lower level functions. These are that:

a) in general, architecture-level requirements are *broadly defined and hard to capture*,

b) the requirements are often *conflicting and multi-dimensional*, and

c) architectural requirements are frequently concerned with the behavior of the system over a *long period of time* rather than at a single instant or short interval.

The process of evaluation for a large, multi-dimensional system is itself complex. Evaluating a balance among a set of multi-dimensional requirements is much harder and less precise than a simple optimization of a variable. The process usually proceeds iteratively, with first designs being subjected to evaluation that leads to revised and refined designs. Thus, the design of large systems like the Internet is inherently an experimental exercise.

It is unlikely that the union of all the features outlined in Section 2 results in an appropriate architecture. One of the objectives of this effort is to validate which goals are essential, and which are best left outside of the architecture. History teaches us that we should be wary of the "second system" syndrome[5].

4.1 The stages of design and evaluation

What is the sequence of steps that would take us from design through preliminary assessment through trial deployment to possible commercialization? In general, there are at least four stages that can be separated, at least conceptually. (We fully recognize that in practice the stages overlap, mingle, and loop back as later experience leads to revised design.)

Stage 1: Mechanism design. In this stage, specific proposals for new mechanisms, protocols and architectural components are brought forward and evaluated. Section 3 catalogs many such ideas that the research community can consider, ranging from traditional networking areas such as addressing and routing to newer areas such as identity and location management. These ideas may be evaluated using simulation or emulation, or perhaps by actual implementation, though it is sometimes hard to test a single mechanisms in isolation. (By analogy, it is usually hard to experiment realistically on a single human organ without its being

[5] According to wikipedia, "the *second-system syndrome* is the tendency when one is designing the successor to a relatively small, elegant, and successful system to become grandiose in one's success and design an elephantine feature-laden monstrosity." See http://en.wikipedia.org.

part of a human body.) What will distinguish this work is strict attention to the full range of architectural requirements, and a process of evaluation (paper analysis and/or experimental implementation and test as appropriate) based on that full set of requirements.

Stage 2: Concept integration. In this stage, some set of proposed mechanisms are selected and pulled together to make up a coherent overall proposal. This "candidate" future Internet would be specified and documented. In this stage it is not implemented, so it cannot be turned on and run. However, it is amenable to evaluation using tools such as those discussed above. (In practice, stages 2 and 3 are entangled, but it is helpful to focus separately on the processes of design and evaluation.)

Stage 3: Preliminary evaluation and assessment. Once a candidate future Internet is proposed, it can be evaluated by a number of means. Some parts of it might be susceptible to partial isolated implementation, or to test via simulation. But at this stage, the process is to a considerable extent an intellectual process carried out on paper.

A first evaluation criterion for any systems architecture is **suitability for purpose** – the initial set of requirements is reviewed to see how the design of the candidate system addresses them. For a complete architecture, this process involves a multi-dimensional consideration of multiple requirements, and the interaction of the various requirements and mechanisms, as discussed above.

A second criterion for any complex system is **modularity of function**. As is well known, modularity allows for such important capabilities as independent implementation, independent evolution of critical system algorithms as system needs change, and independent evolution of different system elements as technology advances. Considering these points together, it becomes apparent that modularity of function is key to one important goal of systems architecture: *multi-generational system lifetime*, or survival of the overall system beyond the lifetime of any particular technology used to build it. Good functional modularity is a classic problem in computer science design.

A third criterion that is becoming increasingly important as networked systems grow in real world importance is **modularity and management of tussle**. In contrast to modularity of function, this criterion is concerned with isolation and management of non-technical concerns, allowing the system to respond effectively to changes in the underlying social and economic environment in which it sits. A simple framing of this idea, often expressed as "separation of mechanism and policy", has been used as a design principle for a number of systems. As our understanding of the interplay between technical systems and the larger world in which they operate matures, a more sophisticated understanding of how to make technical designs resilient to changing economic and social forces[6] can be expected to emerge. This criterion can be evaluated using methods such as stakeholder analysis, as discussed in Section 4.2, to enforce rigorous thinking.

After this stage 3 evaluation and assessment, we will have a candidate proposal for a future Internet that is as good as possible, based on critical thinking, formal tools for analysis, and other structured approaches to assessment. But it is not a running system, and we have no

[6] Or, alternatively, robust and resistant to particular societal pressures, should that be the intent of the designer.

actual experience with it. For this, we must build it, and this is the step in the process where GENI will make the critical difference.

Stage 4: Trial implementation. No system makes it into production without first being implemented and tried. So a necessary step on the way to success is to build the system and see how it works in practice. This is the stage where GENI makes progress possible. Without the option of building, there is no real expectation of deployment. Some other party might be so excited by the paper design in stage 3 that they proceed with a full-scale deployment, but this outcome is not likely. And since few researchers are willing to commit their career to the design of a system that will never be built, a reasonable chance of undertaking stage 4 is necessary if people are going to undertake the earlier stages.

Figure 1 illustrates the power of GENI to transform the research cycle. Without a facility such as GENI, most research is limited to the stage of simulation and emulation, which allows an algorithm or mechanism to be evaluated relative to a simplified model of the real world. GENI allows the research community to go to the next, pre-deployment stage, and experiment with a new idea "at scale", in the context of the real world and its variability.

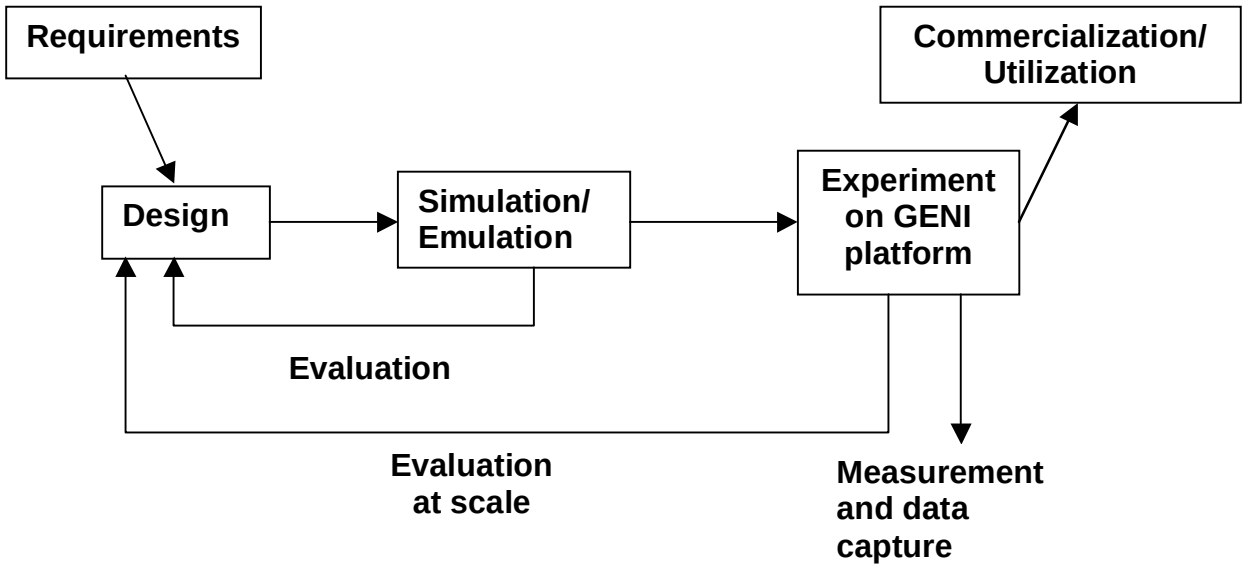

Figure 1

But the picture tells only part of the story. What does it mean to "evaluate" an architecture? A real world test allows a very general process of measurement, assessment, and evaluation. Some aspects of an architecture are amenable to a numerical measurement—for example performance metrics such as link loading. Certain aspects of security can be quantified, such as percentage of down-time[7].

[7] It is important to remember that these numbers can be gathered in two contexts, a controlled and repeatable but perhaps simplified and unrealistic context, or a real world context which gives more

But evaluating an architecture involves much more than quantitative measurements. There are many sorts of evaluations and assessments that must be done. Section 2 proposed a set of requirements for a future Internet, including better security, better manageability, better support for mobility and wireless, a more healthy industrial structure, and so on. Very few of these requirements are amenable to a quantitative measure. How, then, do we judge our system against them during stage 4, when we have a prototype running system?

4.2 Strategies for evaluation

Several strategies are available for evaluation of architectural experiments. Each strategy has both strengths and limitations, and strategies are often best applied in combination.

Strategy 1: Deployment and observation under realistic conditions

Many aspects of a system architecture, particularly those that are performance related, can be evaluated by observation under realistic conditions. Realistic conditions, including the existence of real users, are valuable in circumstances where the characteristics of the operating environment are complex, poorly understood, or difficult to model. The primary limitations of this strategy are that it is not well suited to evaluating aspects of an architecture that are related to change and evolution over time, and that it is not well suited to systematically and rigorously stressing the architecture in any objective way.

One of the benefits of GENI will be consistent instrumentation. While the Internet can be used as an experimental platform for certain classes of systems, one of the drawbacks is that it is often not possible to measure traffic flows and other aspects of usage. Indeed, one of the real frustrations of the research community is that as the Internet became a commercial success, it became essentially impossible to monitor what is happening there. GENI will provide a platform with a rich set of tools for measurement and monitoring. The GENI infrastructure offers full observability of the experiment and the related data – a "God's Eye view". The key benefit of such an experiment is that the researcher can obtain data not visible in real life, where it may be hidden by administrative, implementation, or privacy concerns.

Strategy 2: Accelerated evolution

The basis of this strategy is to stress the architecture by speculating about the world 10 or 15 years into the future, and trying to simulate the conditions that will prevail at that time. This might include higher speeds in the core, a wider range of device capabilities at the edge, wider variation in performance in different system elements, large mobile networks, or shifts in usage caused by increasing security concerns. This future-oriented exploration of the operating space can reveal limitations of the architecture that are extremely unlikely to be seen in normal use.

The primary limitations of this strategy are that it may be difficult to predict the future in enough detail, and it may be difficult to simulate some of the anticipated operating conditions, such as 100 gb/s or 1 tb/s links, 100 million online cars or 100 billion sensors.

Strategy 3: Building on top of it

---

confidence about actual performance but less confidence about repeatability. GENI will let us do both sorts of experiments.

One way to evaluate a system is to **use** it. The Internet is a platform for applications and services. It is a platform for innovation. So a major consideration in judging a new Internet is whether its features are actually useful to developers of higher-level services. Consider, for example, the proposal that a future Internet should have a general set of mechanisms (an architecture, if you will) to manage the concept of physical location and the building of location-aware applications. As the old saying goes: "the proof of the pudding is the eating"—there is no surer way to confirm that the mechanisms are useful than to use them. But this requires that the supporting services actually exist in a usable form. It is very hard to convince an application designer to build an application for a platform that does not actually exist. So part of the motivation for GENI is to allow a candidate for a future Internet to be deployed in a running state, so that we can attract application builders to come and build.

Strategy 4: Intentional perturbation

One important test of a system such as an Internet is the "stress test", where we deliberately push the system to its limits, we subject the system to a range of failures and outages, and we measure and assess its resilience and dynamics. It is not possible to break parts of the real Internet on demand. A facility such as GENI is necessary to perform this class of experiment. **Active experiments** alter the experimental environment in ways that are not likely or achievable in the real world, in order to provide worst case analysis, failure mode evaluation, and similar forces. Such experiments are critical to implementing the "accelerated evolution" strategy outlined above.

A point to consider in designing such experiments is that architectural evaluation is multidimensional. This creates the opportunity for experiments that are explicitly designed to incorporate "real world" behaviors in certain dimensions (e.g. real users) while adopting active set piece behaviors in other dimensions (e.g. economic forces or fault injection). Well-reasoned application of this strategy can lead to quick understanding of architectural robustness or fragility in different key dimensions.

Strategy 5: Stakeholder analysis

Stakeholder analysis should be performed based on the initial requirements, but it should also be performed with respect to any given proposal, since a particular design can create new stakeholders, or shift the balance of power among them. Given an explicit stakeholder analysis of the architecture being evaluated, it may be possible to devise, as realistically as possible, an experiment that includes the behavior of each of these stakeholders. One of the features of GENI that will facilitate these sorts of experiments is to create different slices that embody the different stakeholders, so that their interactions can be role-played in the experiment.

Strategy 6: Integration of experimental and analysis tools

Architectural analysis tools provide a strategy for evaluating architecture prior to implementation, and as well a structured way to guide the experimental evaluation of a prototype running on a facility such as GENI. This new but increasingly productive field of research represents the transition of systems architecture from pure art to a more formalized and rigorous approach. Within certain domains of interest, these tools have advanced to the point where they can offer useful insight into the long-term behavior and properties of an architecture – including those related to system evolution and non-technical (societal and economic) factors.

- **Real Options**. This technique, developed for evaluation of financial alternatives, has been applied to evaluate different architectural choices and help quantify the economic value of different network, protocol, and service architectures.
- **Dual decomposition optimization**. This approach has been used to both derive and evaluate the modularity, robustness, and optimality of architectures for Internet and wireless networking protocols. It provides an increasingly promising framework for reasoning about system modularity and the costs of decentralization in a formal manner.
- **Game Theory**. Game theory is becoming well known in the networking community as a framework for modeling and evaluating architectures that incorporate distributed control by players with potentially differing interests.

Architectural analysis tools offer the potential to also be synthesis tools. The further development of this capability – the ability to reason about systems architecture in a more rigorous manner, and validate that reasoning through experimental experience – represents an extraordinarily promising new paradigm for architecture research and design.

The use of such architectural analysis tools may be helpful in different experimental situations. One valuable role of such tools is to help establish likely worst case scenarios and situations, so that experimental evaluation can be concentrated in areas where it will be most revealing. Another possible role is to establish theoretically grounded baseline performance bounds, allowing the performance of a complex but practical system to be compared against a potentially simplified but well understood model. In this situation, significant deviations in performance between the practical system and the analytic model serve to suggest places where further attention to understanding the behavior of the experimental system might be most useful.

After the experimentation, testing and gathering of experience during Stage 4, several things may happen. Realistically, the design process is not linear. There are loops back to the earlier stages of design, integration and evaluation. There is a chance for successful ideas to move forward to a possible **Stage 5**, a transition to commercial uptake and operational deployment. The process will mature, move forward and backward, always using GENI as the platform to bridge the gap between design and demonstration, and as the platform for critical experimentation and evaluation. NSF has set a goal for the FIND project to conceive of an Internet for 10 or 15 years in the future. This is a long-term effort, which calls for a long-term facility.

4.3 GENI's place in the experimental process

From this discussion of design and evaluation it is possible to summarize why GENI is critical to the success of these ventures, which we call *innovation in the large*.

- GENI provides the platform that fills the gap between paper designs and simulations (the process through stage 3) and the potential to achieve real deployment (after stage 4). There is no real potential for deployment without preliminary implementation and testing. GENI, by filling that gap, empowers and motivates the research community to take up innovation in the large.
- GENI can be used to test initial proposals for mechanisms, protocols and architecture building blocks.

- GENI can be used to perform methodical experiments and controlled experiments on running code that embodies the design of a candidate future Internet or other large distributed system.
- GENI will give us experience in linking emerging tools for architectural analysis to actual experiments on running systems. It will carry us forward in the space called science of design.
- GENI will give us a platform to attack key *grand challenges*, as discussed in the next section.
- GENI is a general platform that can allow us to try a number of competing and evolving ideas during a long-term program of research.

4.4 Beyond a future Internet

The previous discussion used the example of a future Internet to illustrate the importance of GENI, and the range of experiments and observations that will be performed on GENI. But GENI is not limited just to work on a future Internet. A wide range of distributed systems can be deployed and evaluated using GENI. For certain classes of applications and services, the Internet of today might be an adequate experimental platform. So long as the experiment only requires a set of end-nodes (e.g. PCs on the net) and the exact function provided by the routers in today's Internet, the experiment is possible. But applications of the modern era do not have this simple pattern. They are dependent on servers and services that are distributed across the network. To demonstrate and evaluate this class of application today requires either access to a pre-existing set of servers, or the deployment of physical machines around the globe. In many cases, these requirements are a barrier to practical research and experiment. GENI will provide a platform for the research community to develop and test this class of system, and will permit it to perform research that that can push the state of the art with respect to the design principles current today.

5 Requirements for GENI

Section 3 contains a collection of experiments that the research community anticipates doing, using GENI as the experimental environment. Both the range of those experiments, as well as the specific needs of individual experiments, define the requirements that GENI must meet. This section summarizes those requirements, describes in general terms the facility we propose in order to meet these requirements, and provides the rationale for some of the key design parameters and tradeoffs of that proposed facility.[8]

GENI is intended to support two general kinds of activities: (1) deploying prototype network systems and applications, and learning from observations of how they behave under real usage, and (2) running controlled experiments to evaluate design, implementation, and engineering choices. These are two very different activities. Classical science equates its experimentation with the latter, but Computer Science also benefits from building and running prototypes, because building something and watching it run helps us to identify implicit assumptions, the need for different functionality, surprising behavior, unexpected limitations, and so on. In this sense, such "experimental systems" work is like constructing a building—engineering principles tell you whether it is a sound design, but you need to build it and use it to decide how well it serves its purpose.

GENI must support both types of activities, and in fact, GENI should make it easy to first perform a sequence of controlled experiments on a new network system, and then subject the system to real user traffic as part of a longer-term deployment study.

5.1 Functional requirements

5.1.1 Multiple simultaneous experiments

The concept behind GENI is that it can be used to test multiple ideas and concepts. The plan is not to pick one winner and then build and operate it, but to build and operate multiple candidates as part of identifying successful ideas. So GENI needs to support multiple experiments.

The goal of GENI is to allow long-running continuous experiments. We want to gather operational experience with new proposals. We want to attract real applications and real users to try out the concepts being developed. If we are to support real applications and real users, the concept of serial reuse (I run for an hour, then you run for an hour), is inadequate. GENI needs to provide the capability for multiple candidate future networks to be brought into experimental operation at the same time, and kept running for weeks or months.

To support multiple, long running experiments, the designers have based GENI on the concept of **slices.** The concept of slices is that the resources of GENI can be divided up among many different researchers in such a way that each can run his own experiment. One approach to slices is **virtualization,** an idea that has been a part of CS research for decades. Virtualization takes a physical resource (e.g. a processor) and creates the illusion that it is multiple processors, identical to the original except that each runs slower. We have experience today in how to virtualize a processor (indeed, there are virtualized routers available as products today), and in

[8] This section does not yet systematically link experiments to requirements to facility specifications. Such a rigorous treatment is in progress. [FAL07]

how to virtualize a fiber link. The concepts necessary to virtualize or otherwise slice a wireless system are less mature, but there are good proposals. So GENI puts these ideas together to build a new class of facility, a virtualized infrastructure for network.

To set a general expectation, we imagine on the order of a thousand researcher projects utilizing GENI.

Controlled Isolation: GENI must support strong isolation between slices so that experiments do not interfere with each other. GENI's isolation mechanisms should be sufficiently robust to make reproducible experiments possible, and to the extent they are not, it should provide enough feedback about what resources a slice actually receives to enable researchers to evaluate the validity of their results. GENI must provide strong containment for experiments that involve the release of security attacks against new defenses. At the same time, GENI must support controlled interconnection of slices to each other and to the current Internet, allowing researchers to build directly on each other's work, and to draw on existing Internet users and resources. Some experiments, such as tests of new inter-region routing protocols that might replace BGP, require that an experimenter build different regions as different slices, and then allow them to connect together to exchange both data and routing information. This implies mechanisms that enable user opt-in and desirable data exchange between slices, while keeping undesired outside factors from interfering with GENI experiments and containing GENI experiments so that they do not adversely affect the rest of the Internet.

- To permit high-performance, multi-slice experiments (e.g. multi-domain routing experiments), GENI must include cross-slice connectivity at throughputs consistent with the backbone capacity of the GENI facility.

5.1.2 Generality

We are at a time in the design cycle for networks where we need to explore alternatives to the current paradigms. We do not need a facility to help us make the current Internet a little better; we need a facility that can help us demonstrate revolutionary alternatives. The Internet, of course, is itself general, in that it can support a wide range of applications. Now we want to build an experimental platform that can support a wide range of future Internets.

Our approach to generality follows this line of reasoning. We understand what the basic building blocks of networks are today, and what they are likely to be in 10 years. In 10 years, long-haul networks will mostly be build out of fiber optic cables, which will be connected together with a variety of optical and electrical processing elements, some derived from what we see in operation today, and some based on equipment now in the laboratory. Edge networks will be a combination of wired and wireless access, with both higher speeds (to support high-end processors) and lower speed and lower cost (to support embedded processors and sensors). Distributed applications will be based on massive processing and storage facilities. We can see the general trajectory of processing and storage (petabytes of storage and massive, multi-core processing.) If we build the GENI facility out of these components, we have a technology baseline that almost any network proposal would expect to build on.

How much generality is required to support the anticipated experiments?

- We must be able to experiment with packet formats that materially differ from those of the Internet. A large number of anticipated experiments, including security, management, routing, congestion control, accounting, inter-ISP interaction, and mobility all suggest the need for a new packet format.

- We must be able to move beyond the paradigm of packet switching and explore other modes for sharing and resource allocation. The idea of highly dynamic management of traffic aggregates implies a need for rapidly reconfigurable optical circuits in the core of the GENI facility. Experiments with high-throughput end-to-end flows imply the ability to support end-to-end circuits (streams of data) that may not include any sort of framing or multiplexing at all.
- We must be able to exploit specific features of the different technologies included in GENI. In particular, the mode of slicing or virtualization that is used for each technology in GENI must reveal any important technology-specific features of the underlying technology, such as wireless broadcast. We must not constrain the GENI experimenter to seeing the lower-level technology only through an intermediate and constraining abstraction, such as the abstraction of a point-to-point link or a time division packet.
- We must be able to experiment with architectures that include network-level operations other than simple packet forwarding. For example, some proposals for enhanced security imply the ability to experiment with nodes that examine and regulate data flows based on content, identity, capabilities or credentials. Low-level dissemination schemes may involve nodes that provide functions such as multicast or any-cast. Nodes may examine, store, or resend data, and reassemble and reformat low level elements such as packets.

Diversity of technology: GENI must include a wide class of networking technologies, spanning the spectrum of wired and wireless technologies available today. GENI must also be extensible—with explicitly defined procedures and system interfaces—making it easy to incorporate additional technologies, including those that *do not exist* today. This will allow GENI to be useful to a broad range of researchers, remain useful over a much longer lifespan, support GENI's role as a low-friction vehicle for deployment of new technologies by both academic researchers and industrial partners, and foster close collaboration between "device researchers" and "systems researchers."

The design of GENI must, of course, balance generality with cost, and this balance will always be a matter of judgment. The designers of GENI have chosen to focus on a class of experiments that depend on operation over a heterogeneous selection of technology, rather than the optimized operation over a particular technology. The designers of GENI have also chosen to focus on experiments that benefit from actual deployment in the real world, as opposed to emulation in a lab.

We understand that there are some very advanced technologies (e.g. quantum networking) that we may not be able to support at a global scale. We have chosen to avoid ridiculously costly options such as launching low-orbiting satellites, which were all the rage a few years ago and may return as an interesting option. We have chosen to concentrate on wireless access at the edge, under the assumption that the future shape of wired access is more predictable, and can be approximated by the wired access to the enterprise desk of today (gigabit access to the workstation.) As we continue to explore the range of experiments that are proposed for GENI, some of these decisions can be re-evaluated.

5.1.3 Support for real applications

The goal of GENI is to allow experimenters to gather operational experience with their ideas, experience that is as close as possible to "real world". There are many requirements that derive

from this high-level goal. One requirement is that GENI should be able to support not just a future network, but also the applications that might run on that network. A network without applications is not actually being used or evaluated; if the only traffic on the network is an artificial test load, that experiment might as well be done in the lab.

But to attract real applications, the GENI must include facilities for development and deployment of applications, not just data transport. Today, the hardware for network level data transport and the hardware for application support is very different. Hardware for data transport is specialized, high-performance packet forwarding equipment, and GENI must provide a generalized version of this class of equipment. But applications tend to run on servers with large primary memory, large disks, and massive amounts of general purpose processing. Unless there is equipment of this class provided in GENI, there is no way that candidate future networks can attract real application builders to build and deploy new applications. Section 5.1.5 discusses the specific requirements to support the anticipated range of application-level experiments.

5.1.4 Support for real users

In turn, if we are to gain real experience with real applications, we must allow real users to try them out, and make real use of them. So the concept for GENI is that it will have the reach and connectivity to allow real users (perhaps members of the CS research community, or perhaps more broadly) to make use of the services and applications that are being evaluated on GENI.

What does it mean to support real users?

- We must provide a means for a sufficient base of users to have access to GENI. This implies that the GENI facility must reach "to the edge" of the network, where the users connect. In practical terms, this means that the GENI facility must include apparatus that is located on the campus of research facilities, with connectivity all the way to the end-node computers used by the target users.
- For some experiments, it may be adequate for users to gain access to GENI experiments by connecting over regions of the Internet. This implies that there must be a rich connectivity between GENI and the Internet of today. But some GENI experiments may imply the use of concepts that are foreign to the current Internet, so that there must be an adequate pool of potential users that have their end-node computers directly connected to, and a part of, the GENI infrastructure.
- Some experiments that involve users will require that their end-nodes be modified. Many experiments, including proposal for security, management, congestion control and mobility require that the end-node use a different protocol stack. Experiments with different schemes for route selection or service provisioning will require the implementation of new control protocols, APIs and user interfaces. So part of the GENI development process must include making it easy for experimenters to install new software in popular operating systems, such as Linux or Windows. Some support for slices needs to be provided for the end-nodes that are attached to GENI.
- Some experiments may require that the users have rich forms of connectivity. For example, the experiment in section 3.2.3 on user-selected routes implies that the users must actually see a choice of routes with materially different characteristics. This

capability can, to some extent, be supported by virtualizing the connection from the end-user to the GENI facility, but there must be adequate diversity in the resulting routes.

- Some experiments may be designed so that users can exploit their services without taking explicit action. For this class of experiment, it would be beneficial if the GENI infrastructure could be involved in the mechanisms by which users join in the experiments, so that the users can be switched out again if the experiment crashes.

5.1.5 Fidelity

GENI should provide an environment that corresponds to what one might expect in a real future network. This means individual components must expose functionality at the right level of abstraction, and it must be possible to arrange these components into a representative network.

Reach: GENI must have as wide a reach as possible. This is necessary to support experimentation at scale, and to maximize the opportunity to attract real users. Access cannot be limited to only those few sites that host backbone nodes. Wide deployment also implies a rich interconnection of the facility to the legacy Internet.

Topology: One important aspect of fidelity is that the topology, physical scale and connectivity should mimic the real world, as we anticipate it being in 10 years, to the extent possible. Here are some important considerations as we evaluate the fidelity of the GENI topology

- Keeping delays within a small factor of physical distance: Commercial ISPs typically try to limit the end-to-end propagation delay for each pair of backbone sites to some small multiplier (e.g., 2X) of the "air miles" between the sites. Otherwise, a competing ISP with a direct link between the same two locations could offer much lower latency. For most transport protocols, achieving high throughput requires low propagation delay, making propagation delay an important consideration even for elastic applications like Web browsing.
- Path diversity: Many experiments with new network architectures capitalize on the presence of multiple paths between a pair of sites; some architectures even need multiple link-disjoint or node-disjoint paths. For example, some architectures perform load balancing by splitting traffic over multiple paths, whereas others switch from one path to another in response to congestion or equipment failures. Some experiments, such as routing based on traffic diffusion, described in section 3.2.3, require a completely connected mesh of nodes, although this can be simulated to some extent by virtualizing the links in the experiment.
- Underlying fiber paths: The existing fiber-optic map in the United States imposes limits on the specific backbone sites that can have a direct fiber-optic connection between them. Placing backbone nodes in the key cities where multiple fiber-optic connections are available is extremely important to reduce the cost and deployment time of GENI. In addition, though it is possible to provide the illusion of dedicated links between any pair of backbone sites, providing links that match the underlying fiber map reduces cost and offers a more realistic deployment scenario.
- Major interconnection points: Deploying GENI backbone elements at existing interconnection points where other ISPs have their backbone sites would allow GENI to amortize the costs of space, power, and "hands and eyes" support. Locating GENI

backbone nodes at major exchange points would be useful for acquiring upstream connectivity to the legacy Internet; similarly, having GENI backbone nodes at major aggregation points (such as the GigaPoPs) would facilitate efficient, low-cost connectivity to edge sites, such as university campuses.

Realism of virtualization: The idea of slices, or more specifically virtualization, is central to the proposed approach to building GENI. However, virtualization may itself be a source of unrealistic behavior, which has to be identified, minimized, and specifically documented. A specific example of an issue is *jitter*, or variation in the timing of slice execution, if the scheme for virtualizing the underlying physical resource involves sharing in the time domain, or "time slicing". If the time slices are large, a program may have a highly variable latency in its actual running time, and this may disrupt some experiments, especially experiments such as those described in section 3.1.6 involving real time applications with tight time bounds.

Physical distribution: A number of the applications and services to be developed and run on GENI will explore different tradeoffs along the spectrum of distribution. GENI must provide a realistic platform to test systems that range from centralized, to distributed on a regional, campus or end-node basis. This requirement implies the need for highly distributed computing facilities in GENI. It would not be acceptable to simulate a highly distributed service platform with slices on a centralized machine, because this simulation would lead to highly unrealistic latencies for traffic, and this lack of fidelity would be unacceptable. For many applications, round trip delay rather than bandwidth determine the overall performance as measured from start to finish of high-level operations. Many applications have a pattern of interaction that involves a series of round trip queries or transactions (as opposed to a single bulk data transfer). A centralized server in the middle of the country might provide round trip latencies of 50 ms., which implies a maximum of 20 interactions per second, no matter how fast the computers or the networks. But moving to a decentralized pattern where the interactions are on a local scale can decrease total running time of such applications by more than 2 orders of magnitude. GENI must provide a highly distributed platform to support these sorts of applications. The lack of a processing platform at any research site would prevent the researchers and users at that site from having a realistic exposure to the full range of distribution.

Scale: Large commercial distributed systems today have tens of thousands of physical nodes at thousands of sites. The Akamai system, for example (one of the larger commercial platforms for distributing content and related activities) is over 20,000 servers on 1,000 networks (many with multiple points of operation), currently in 71 countries. GENI must give the research community the ability to build advanced, experimental systems that are at least in the same league as what the commercial world has today, perhaps to within an order of magnitude of currently deployed distributed systems.

Failure modes: There are two sorts of failures to be considered in GENI. The first is the intentionally induced failure of a virtual component to observe the consequences on the running system. It should not be difficult to design the virtual components so they can fail on command, but this mode of testing limits the experimenter to the class of failures that he has pre-conceived. The second sort of failure is the unanticipated failure that is injected into an experiment by the failure of some real, underlying component. These sorts of failures are very valuable, since they take an unanticipated form and stress the experiment in unanticipated ways. However, unless care is taken, real failures in the GENI facilities might lead to very unrealistic failures of an experiment. For example, if a highly sliced node fails, this will result in

the simultaneous failure of a node in each of the supported slices, which might be a very unrealistic degree of correlation (or perhaps not, if it mimics a massive regional disaster.) If a number of virtual links are derived from a single physical link, then the failure of this physical link will cause the simultaneous failure of all the virtual links, which makes it difficult to demonstrate systems that are resilient in the face of a limited number of link failures.

The ability to inject failures into the system, both individual and massive, is particularly important for the challenge question concerning service in times of disaster, where we need to mimic the consequence of a large scale disaster on a running system.

5.1.6 Support for all aspects of a new network architecture

When thinking about a new network architecture, it is easy to concentrate on the aspects that relate to actual data forwarding. However, most of the issues and requirements described in Section 2 relate to other aspects of the design, such as security and management, and these requirements require us to attend to all aspects of virtualization of the underlying technology.

Support for management: It is important that the management aspects of all devices be fully virtualized. Each virtual device created as a slice of a physical device must present a full management interface within the slice, and if the device has operating modes or states, these must be separately settable for each slice. It must be possible to bring up and shut down a slice of a component, and if a device has a physical management interface that is used by system operators, this too must be virtualized to the extent possible, so that we can have the equivalent of "virtual system operators", who play the role of operator within any one slice. Real failures must manifest as realistic failures of the virtual element to the extent possible, for example signals about estimated time to repair.

Support for security: The desire to support experiments in enhanced security has several implications. First, the GENI infrastructure itself must be stable and secure to an adequate degree, so that that experiments that claim enhanced security or availability can actually demonstrate these virtues. Second, the mechanisms for isolation among slices must be very robust, so that an experiment that involved an attack on a system in one slice cannot "escape" and attack other experiments. Third, there may be a requirement for specialized security technology, such as hardware-specific unforgeable identity tags, key generators, or physical hardware interfaces for secure management.

Support for anticipated future capabilities: As we envision the facilities that should be included in GENI, we must remember that in 10 years, there may be features that will be commonplace then, but are not yet realized in any effective way. There are obvious issues of performance, but beyond this there are many other functional and operational issues. For example, in 10 years every device, no matter how small, may always know where it is. Devices may have special physical interfaces that are used for secure management and configuration. Devices may have new modes of powering, which allow them to remain up when the normal power goes down. Mobile devices, and as well experiments that attempt to demonstrate a highly resilient network suitable for operation in times of disaster, will be very concerned with issues of power, and we must consider how to virtualize a source of power so that different experiments cannot deplete each other's power allocation.

5.1.7 Support for experimenters

Ease of Use: GENI must remove as many practical barriers as possible to researchers being able to make full use of the facility. A small network or distributed systems research project might be

conducted by a single principal investigator along with a single student. For GENI to be practical for these users, the overhead of understanding how to map their intended experiment onto GENI must be within reach. This means GENI needs to provide a rich set of tools for configuring, monitoring, and debugging experiments, a rich set of common utilities to be used by experimenters, and predictable and repeatable behavior for experiments running on the system. At the same time, GENI will also need to provide access to the full set of capabilities of the system for "power users."

Observability: GENI must offer strong support for measurement-based quantitative research. In GENI, measurements are required for researchers to debug, understand, evaluate, and demonstrate their new network protocols or architectures in an operational GENI setting. Measurements are important inputs for modelers developing mathematical or simulation models, who want to answer "what if" questions beyond the specific GENI setting. Measurements are also useful for those seeking to establish the theoretical foundations of new protocols or architectures and who wish to validate assumptions underlying their models. On-line measurements will also be needed by a new generation of network/system/service management capabilities developed on GENI, which allow real-time control/configuration based on observed network events. Indeed, measurement capabilities will be a crucial component of GENI to nearly every researcher.

This requirement means that the GENI resources, along with all the network systems deployed on it, must be heavily instrumented. The generated data must be collected and archived, and analysis tools developed.

A successful measurement strategy will require several questions and issues be addressed:

- To what extent can common network instrumentation/measurement capabilities be shared among users? To what extent should measurement and monitoring be done within a slice (which is more representative of eventual operating conditions, but more work for the experimenter) as opposed to being done in the GENI infrastructure?
- What aspects of GENI must be instrumented? Should GENI include special components for capture and processing of data? (One example might be a high-speed probe attached to a fiber that can capture data a some level going across the fiber.) If such devices are contemplated, should they be virtualized and made available "inside" the slice, or as part of GENI itself?
- How does GENI address issues of privacy? How is captured data archived and used? Under what circumstances can the larger research community see data gathered as part of a specific experiment? For example, experimental systems that offer privacy or anonymity to experimental users must not have these guarantees compromised arbitrarily by the GENI facility itself.

Fail-safe: GENI must be secure, so that its resources cannot accidentally or maliciously be used to attack today's Internet. To this end, GENI should be designed to operate in a "do no harm" posture: an experiment should run within a "bounding box" that limits what it can do; it must be possible to trace network activity back to the responsible experiment (and experimenter), so that any problems or complaints can be addressed; and should GENI enter a period where activities of some components cannot be adequately monitored or controlled, GENI should

restrict those activities by other means to a point where safety can be assured (e.g., by shutting down a slice or bringing GENI as a whole into a safe state).

Sources of real traffic: One of the most important characteristics of GENI is that it will allow prototype systems to tested with real traffic. This involves two issues: providing the technical means by which users could direct their traffic over GENI, and giving users a reason to do so. The former issue was discussed in 5.1.4, and GENI is expressly designed to provide the kinds of support listed there. The second issue is subtler. Some prototype systems will offer desirable services that users will seek out, and these systems will have no trouble attracting users. For example, the Coral, CoBlitz, and CoDeeN content distribution systems [FRE04, PAR06, WAN04] currently deployed on Planetlab have attracted a large user base, carrying 4TB of traffic and communicating with 1M unique Internet hosts every day. However, there are other designs whose benefits are less obvious to the user and are unlikely to attract a significant user community if deployed on their own. For instance, systems providing better manageability or more scalable routing might not provide users any tangible benefit in the short term, even though these designs would be of tremendous value in the long term.

GENI must provide a way such experiments can be run with real traffic. One approach would be to have several large-scale popular services, such as content distribution networks, running on GENI with large user populations. Prototypes for underlying architectures could be slid underneath these systems so that they could gain experience with real traffic. In some cases, it will be possible to have several such prototypes supporting a single high-level service; this will require partitioning at the application level.

Thus, one important goal for GENI's management is that popular services are developed (either through the natural course of experimentation, or through explicit design) that can provide large sources of real traffic for designs that operate below the application level.

5.1.8 Federation & Sustainability

GENI must be designed for a 15-20 year lifetime, going well beyond a 5-7 year construction phase. To ensure the sustainability, it should be possible for participating institutions (including countries) to contribute resources in return for access to the resources of the GENI as a whole. It should also be possible for new research communities to "opt-in" by connecting their purpose-built networks (including dedicated transmission pipes and sensor networks) into GENI and running their applications and services in a slice of GENI. Both of these scenarios imply the need to support *federation*. In addition, GENI must be designed with *operational costs* in mind, including hardware upgrades, software maintenance, and ongoing operational support.

Addition of new technology: To permit upgrades and to take into account new technology innovations, it must be possible to add new technology to GENI while it is in operation. Examples might include new, more advanced optical switching technology, novel wireless technology, or new means to determine location or other operating conditions. This requirement implies open hardware interfaces, but also the ability to virtualize these devices, and the ability to incorporate new devices into the GENI management mechanisms easily.

Living in the future: In the specification of GENI, especially as we balance cost with function, it is important to remember that GENI is supposed to be a tolerably realistic emulation of a networking technology base 10 years in the future. In 10 years, if Moore's law holds, we can expect the cost-performance tradeoff to improve by a factor of 100. This implies that what we build today may cost 100 times as much as an operational system 10 years from now. As we size

components, we should not expect the cost of GENI to be measured against the cost of an operational system today. In the past, the simple measure of performance has been link capacity, but in GENI, performance may equally be measured in the computing and storage capacity of the processing nodes. As well, it may be necessary to expend resources to mock up (and then virtualize) certain specific features that we anticipate will be commonplace in 10 years, such as the ability of every node to know where it is physically, for mobile nodes to have greater processing power or novel display modalities, or for every physical object in the world to have attached to it a link to its cyber-equivalent.

5.1.9 Striking a balance

While it would be difficult to argue against any one of these requirements in isolation, what makes GENI a unique and compelling instrument is how it balances these requirements to support research that simply cannot be done today. This balancing act has two aspects. First, it involves resolving conflicts among requirements; these tensions are discussed in Section 5.3. Second, it involves recognizing the specific combination of capabilities that are unique to GENI—capabilities that are not available in a more limited facility (e.g., in a single researcher's lab or a smaller more special-purpose testbed). They include: (1) wide-spread deployment, (2) a diverse and extensible collection of network technologies, and (3) support for real user traffic. These three properties effectively define GENI's value proposition.

5.2 A reference implementation for GENI

This section gives a high-level overview of the GENI facility, designed to support the research outlined throughout this report. For a more detailed description, the reader is referred to the full design document [DESIGN].

At the lowest level, GENI comprises a *physical network substrate* that includes a diverse collection of network devices, communication links, and access networks. Each experiment using GENI will run on some subset of the resources in the GENI substrate. We call the substrate resources bound to a particular experiment a *slice*. Each slice will include some number of nodes (including both physical processors and virtual machines multiplexed on shared hardware) connected by links (including both physical links and virtual links), and spanning some number of network types (including wired, wireless, and sensor networks).

The GENI facility will include a *global management framework* that allocates resources to slices, ensure that slices do not interfere with each other, and help researchers manage their experiments. The management framework will support two different usage models for slices. In the first model, researchers with short-term experiments will acquire a slice of GENI resources for a limited period of time, run their experiments, and release the GENI resources so they are available to other researchers. In the second model, researchers that wish to deploy and evaluate long-running services that support a live client community will acquire a slice of GENI resources for an indefinite period of time. This implies that GENI must support multiple concurrent slices; it is not sufficient to "time share" GENI resources over course-grained time intervals.

5.2.1 Physical Network Substrate

The physical network substrate consists of an expandable collection of building block *components*. Although no single component could do so by itself, the set of components chosen

for inclusion within GENI at any given time are intended to allow the creation of virtual networks covering the full range needed by GENI's constituent research communities.

We expect the set of building block components to evolve over time as technology and research requirements advance, but the GENI execution plan defines an initial set of components to be deployed:

Programmable Edge Clusters intended to provide the computational resources needed to build wide-area services and applications, as well as initial implementation of new network elements.

Programmable Core Nodes intended to implement core network data processing functions for high-speed, high volume traffic flows.

Programmable Edge Nodes intended to implement data forwarding functionality at the boundary between access networks and a high-speed backbone.

Programmable Wireless Nodes intended to implement proxies and other forwarding functionality at the boundary between wireless and wired networks.

Mobile Client Devices intended to run applications that give end-users access to experimental services available on the combined wired/wireless substrate.

A **National Fiber Facility** intended to provide 10Gbps or higher light path interconnection between GENI core nodes, forming a nationwide backbone network.

A large number of **tail circuits** of varying technologies, intended to connect GENI edge sites to the GENI core, and the GENI core to the current commodity Internet.

Multiple **Internet Exchange Points** connecting the nationwide backbone to the commodity Internet.

One or more **Urban 802.11-based Mesh Wireless Subnets** intended to provide real-world experimental support for ad-hoc and mesh network research based on an emerging generation of short-range radios.

One or more **Wide-Area Suburban 3G/WiMax-based Wireless Subnets** providing open-access 3G/WiMax radios for wide area coverage, along with short-range 802.11 class radios for hotspot and hybrid service models.

One or more **Cognitive Radio Subnets** intended to support experimental development and validation of emerging spectrum allocation, access, and negotiation models.

One or more **Application-Specific Sensor Subnets** capable of supporting research on both underlying protocols and specific applications of sensor networks.

One or more **Emulation Grids** that support controlled experiments by allowing researchers to introduce and utilize artificially generated traffic and network conditions within an experimental framework.

5.2.2 Global Management Framework

The second major part of GENI, the global management framework, knits the building block components together into a coherent scientific instrument—a single global-scale facility that is capable of supporting the research cycle outlined in this document. The management framework, which is primarily implemented in software, is responsible for embedding slices into the GENI substrate, and controlling these slices on behalf of experimenters.

An important attribute of the management framework is its support for decentralized control. Individual building blocks are largely autonomous and self-managing, but can be included in a slice by invoking a well-defined interface. Collections of building blocks—e.g., complete wireless subnets, regional subsets of the edge sites, the composition of components that form the backbone—can be treated as aggregates and managed independent of each other. Similarly, outside organizations that contribute their own resources can federate with GENI, while retaining autonomous control over their components. This framework also allows for a rich set of management services to be developed independent of each other, with each service providing a unique set of capabilities to a specific user base. All of these independent management elements are presented to researchers as a single logical entity, through the use of a unified web interface, yet the underlying management framework is designed to support autonomous and decentralized control.

The key to the management framework is to cleanly separate a minimal and stable core from an extensible set of high-level management services. This minimal core—which we call the *GENI Management Core* (GMC)—forms the ``narrow waist'' of the GENI architecture. It logically connects a diverse and ever-changing set of building block components with a rich and evolving set of management services. It is the management services that assist users as they embed slices into the substrate, and control those experiments as they run. Lowering the barrier-to-entry for researchers that want to use GENI's physical substrate is the main objective of the management services.

5.3 Tensions

Many of the requirements outlined in the previous section are synergistic. For example, a wide-spread deployment naturally supports greater user access, and making GENI extensible (so it can accommodate new technologies) is consistent with its support for federation (so new communities and partners can add their resources to GENI).

On the other hand, there are intrinsic tensions among some of these requirements, as well as between different types of experiments that value the requirements differently. This section identifies several of these tensions, and offers guidance as to how conflicts should be resolved.

5.3.1 Sliceability vs Fidelity

Balancing sliceability and fidelity is one of the most fundamental challenges facing GENI. On the one hand, virtualizing the underlying hardware allows many researchers to share a common set of resources, and can increase flexibility by synthesizing multiple and/or higher-function virtual environments from a single physical resource. On the other hand, virtualization has two potential limits: (1) it allows for the possibility that one experiment might interfere with another experiment, and (2) it potentially hides certain capabilities and properties of the underlying hardware. Both give the facility less fidelity than if a researcher had the resources all to him or herself. Note that virtualization does *not* imply that all slices equally share the available resources, and hence, subjected to unpredictable performance. An admission control mechanism can be used to limit the number of active slices at any given time, and resource guarantees can be made to certain slices. Still the possibility of interference exists.

On the surface, this particular conflict is easy to resolve—GENI should provide strong isolation between slices and the lowest level of virtualization that the technology allows. Any given component may not provide the desired level on day one, but advancing the state-of-art in virtualization over GENI's lifetime is an ongoing objective. Note that higher levels of

abstraction should also be retained for those experiments that do not want to be exposed to low-level details, but virtualization should be pushed as "low" as technically possible (cost allowing).

However, there will be those that argue that any amount of virtualization is too much, and that their research requires access to "bare metal." This might be because of the need for access to a component-specific feature, or because virtualization introduces too much unpredictability in timing measurements. There may also be resources that simply cannot be virtualized. GENI does not preclude the possibility that dedicated hardware elements can be allocated to some slices. There are two obvious ways to allocate physical devices to specific experiments, a technique we call *partitioning*. Each has its drawback.

Resources can be shared in time, allocated first to one experiment and then another. This approach means that the resource cannot sustain a real user workload, and hence limits its appropriateness for deployment studies. Some fraction of GENI's resources can be shared in this way, as long as sufficient capacity is available to support deployment studies. (As noted above, even when virtualization is employed, an admission control mechanism may be used to limit the number of slices that can be active any given time, analogous to time-based partitioning of resources.)

Resources can be physically replicated. This approach would mean that only a limited number of researchers can include a given resource in their slice. This may be necessary for certain high-cost resources that cannot be easily virtualized, in which case it will be necessary for the community to either prioritize their research or find ways to synthesize their many experimental systems into a few comprehensive systems. While we might imagine a thousand researchers sharing GENI as a whole, we might see perhaps only tens of research projects sharing access to any high-cost/non-virtualizable resource in this way.

Independent of the technique used to slice resources, a GENI policy committee will necessarily be involved in prioritizing resource allocation decisions.

5.3.2 Generality vs Fidelity

Designing GENI to be general (programmable) is potentially at odds with perfect fidelity. For example, a researcher could argue that to faithfully evaluate a new function or protocol it is necessary experiment with a commercial implementation, or possibly with a function-specific hardware implementation. In practice, however, such an implementation is likely to expose a limited interface rather than be generally programmable. Such a device has perfect fidelity for a narrow set of experiments, but less value to the larger research community. On the other hand, an open source, software-based implementation of the same function or protocol might run on a general-purpose component that other experimenters can share, but without the performance or fidelity of the special-purpose implementation.

Clearly, it should be possible to make a merit-based case for the special-purpose component that benefits a narrow set of researchers, but it is generally expected that some amount of fidelity will be sacrificed to support a general-purpose facility that serves a wide-range of research. We also note that more narrowly defined communities should be allowed to connect their special-purpose components to GENI, and make them available to interested researchers.

Related to the issue of generality versus fidelity is the issue of simplicity: researchers want to work at a low enough level of abstraction so that important system details are not hidden, but at

the same time, they do not want to work at such a low level that they have to reinvent uninteresting (to them) layers of software just to create an environment that allows them to address their specific research problem. This is actually a unique opportunity for GENI—it should support multiple levels of abstraction, and over time, build up a suite of shared code for commonly used functions. Researchers should be able to work at whatever level of abstraction best matches their needs.

5.3.3 Architectural Design vs Technology Development

We expect an on-going tension between researchers wanting to use GENI to test and evaluate new networking technologies, and those wanting to evaluate new architectural designs that (among other things) take the capabilities of new technologies into account. The former tend to focus on single components, while the latter must take a more comprehensive (end-to-end) perspective. GENI's policies should favor architectural research (broadly defined) that takes advantage of the fact that it spans a diverse collection of hardware resources. This is because no individual technology is fully validated until it has been shown to work with real users in a given context, but also because we are interested in exploring alternative architectures that are capable of integrating a diverse set of technologies.

We note, however, that there is value to component developers being able to evaluate their technology in the context of end-to-end architectures and under the realistic workloads GENI is expected to generate. GENI should allow such technologies to be plugged into the facility once they are mature enough to support GENI users, but we expect early-stage technology development (both hardware and software) to happen outside of GENI. (There is also likely to be a transition path whereby a new technology is made available to early adopters in a subset of GENI.) To make a case for adding a new component to GENI, it will need to support the interfaces defined by the management framework, be sufficiently programmable to give researchers the flexibility they need, and to the extent possible (see the above discussion), be sharable by multiple slices.

Note that this discussion does not directly address the question of what technologies are initially included in GENI. This decision is driven largely by the requirements of the specific research to be conducted on GENI. In general, however, we observe that the overriding goal is to include a diversity of technologies that stress the "corner cases" of comprehensive network architectures.

5.3.4 Performance vs Function

A question often asked about a network is "how fast does it go?" Asking this question of GENI raises the question of performance goals within GENI's design. In the past, performance-related objectives have often defined network testbeds, with speed becoming the key measure of success.

In contrast, GENI's research objectives are broad, and its success metrics focus on properties other than speed. As a result, GENI's design is not focused on performance, and in fact many of the mechanisms used within GENI dramatically increase the challenge of achieving high performance. Despite this, performance cannot be neglected; if GENI does not offer sufficient performance to be useful, it will not be used.

Unfortunately, performance is not a single metric. Rather "performance" encompasses a number of metrics, considered along at least three dimensions. Each of these dimensions affects a different class of experimenters and users of GENI:

Relative performance is the ratio of performance at one point in the network to performance at other points, or of one performance metric to another performance metric at some point in the net. Relative performance ratios may have a strong effect on network architecture, as well as determining the types of operations that can be performed on data within a network.

Absolute aggregate performance is the level of performance available to meet overall system demand at any given place and time. Absolute aggregate performance is important to supporting applications such as content distribution and flash crowd management.

Absolute single-flow performance is the level of performance available to a single application session. Absolute single-flow performance is important to supporting new high-demand applications, such as HDTV video or 3-D data visualization.

In each of these dimensions, there is tension between performance, function, and cost. This tension is strengthened by GENI's objective of providing programmable and sliceable substrate across a range of technologies. Performance levels that are simple to reach in a tuned, fixed-function component are often expensive or difficult to attain within a more general-purpose, flexible system element. Further, reasoning about GENI performance metrics is made difficult because GENI's performance objective is the more nebulous "good enough to meet GENI's research support goals", rather than a simpler, more specific one such as "as fast as possible" or "100 Gbps."

A final tradeoff related to GENI performance concerns how the system evolves over time. It is clear that performance levels sufficient for the first phase of GENI deployed in the near future will be insufficient for the lifetime of the facility. For this reason, performance goals in the near term must be related to longer-term plans for ongoing upgrade and improvement of the facility.

5.3.5 Scale vs Ease of Deployment

Scale is one of the main motivators for GENI. Currently, researchers can easily set up small wired testbeds in their labs, but cannot experiment with their designs at scale with real user traffic. GENI would provide a way for them to do so.

However, the story is quite different with wireless. Radios, unlike wires, are considerably more complex in their propagation and interference, and wireless network protocols have to cope with many more vagaries than their wired counterparts. This complexity makes it much harder to develop sound testbeds to run scientifically meaningful experiments. The sheer effort required to develop and run a wireless network testbed at moderate scale is daunting enough that very few groups around the country (and world) have managed to do it with any degree of consistency.

Put another way, wired networks are generally well-understood at small scale, it is only at large scale when one has to combat the richness and diversity of the "wild" Internet that existing research tools are not sufficient; hence, GENI focuses on redressing this weakness with a facility incorporating both scale and diversity. In stark contrast, wireless networks are poorly understood *even at small scale* (to wit, the theoretical network capacity of even a three-node radio network still remains an open question!) and the underlying simulation and modeling tools do not predict actual behavior or performance. This shortcoming is not for want of effort – much

work has been done on developing more accurate simulators and models – but because of the sheer complexity and number of "degrees of freedom" in wireless networks (e.g., noise, interference, obstacles, movement, transmit power, antenna radiation patterns, antenna imperfections, adaptive modulation and rate adaptation, dynamic topologies, etc.). Each one of the items mentioned in these parentheses is either a non-existent factor or has been tamed on wired networks.

By providing prefabricated *kits* for moderate-sized wireless deployments, GENI can greatly reduce this deployment barrier. This would allow wireless network researchers at different institutions to develop, try out, and refine their ideas in settings where the results are much more believable than ever before. Moreover, since GENI's design allows easy federation, these kits can be connected into GENI to add to its aggregate scale and diversity.

5.3.6 Networking vs Applications Research

GENI is neutral about what level of the network researchers focus their efforts, and so does not draw sharp lines between network low-level protocols, high-level network services, and end-user applications. Any research that benefits from wide-spread deployment, diverse network technologies, and support for realistic network conditions should be supported.

The critical point-of-tension is that GENI is designed to support research in networking and distributed systems—as opposed to simply providing bandwidth to end users—yet it also benefits from traffic generated by real users. It will be necessary to evaluate the research value of traffic generated by a given slice to decide if allocating resources to that slice is warranted, rather than merely providing an infrastructure service to some user community. We can imagine three ways in which a research group justifies the value of traffic they are carrying: (1) by making traffic traces available to other researchers, (2) by providing a novel network service whose efficacy needs to be evaluated, and (3) by offering to run as part of (on top of) a novel network architecture.

Note that new communities that find value in some capability of GENI—or some innovative service deployed on GENI—are free to augment GENI with enough capacity to carry their user traffic, independent of other research considerations.

5.3.7 Design Studies vs Measurement Studies

GENI is being designed primarily to allow researchers to experiment with new network architectures and services not available today, and this purpose will be the primary factor used to prioritize among various design choices and resource allocation decisions. Our hope and intention, however, is that the GENI facility will also provide a new capability for monitoring the current Internet. We believe such dual-use is possible because both capabilities require wide deployment, rich interconnection to the existing Internet, and heavy instrumentation. Using the GENI facility as a platform to monitor the current Internet is a secondary goal that will also inform its design.

5.3.8 Deployment Studies vs Controlled Experiments

We do not view the two primary usage models as being in conflict—a research group might naturally progress from a series of controlled experiments to a long-term deployment study—but there is an important difference in how the two models stress the facility. Both are related to security.

A controlled experiment attempts to both eliminate all outside (uncontrolled) influences from affecting the experiment, and keep the experiment from impacting the rest of the world. The latter requires strong *containment* mechanisms, so that for example, an experiment that measures the effectiveness of a new malware-prevention architecture is not allowed to escape onto the Internet. Because such a breach of containment could have a catastrophic effect, it is likely that experiments will need to be reviewed to evaluate such risks.

In contrast, a deployment study necessarily involves an experimental service interacting with real users, including both individuals that are trying to abuse the network in some way, and individuals that are trying to use the network to transport illegal content. GENI must be willing to carry such traffic; it cannot be isolated for the sake of security. Thus, GENI is expected to behave like an ISP in today's Internet in that it must be responsive to complaints when they are raised. This means it must include auditing mechanisms that allow operators to identify badly behaving experiments, so that they can be quickly isolated or shut down. In general, it must be possible to rapidly bring the facility as a whole into a safe and controlled state.

6 References

[AKE04] Akella, A., Pang, J., Maggs, B., Seshan, S. and Shaikh, A. "A comparison of overlay routing and multihoming route control." *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*. 93-106. 2004

[AND03a] Anderson, R., "Cryptography and competition policy: Issues with trusted computing." in *Workshop on Economics and Info. Sec*, (2003), 1-11

[AND03b] Anderson, T., Roscoe, T. and Wetherall, D., "Preventing Internet denial-of-service with capabilities." in *ACM Hotnets Workshop*, (2003)

[AND05] Anderson, T., Peterson, L., Shenker, S., Turner, T., Editors. "Overcoming Barriers to Disruptive Innovation in Networking." *Report of NSF Workshop*, 2005

[APP04] Appenzeller, G., Keslassy, I. and McKeown, N. "Sizing Router Buffers." *ACM SIGCOMM 2004*, Portland, 2004

[AUM02] Aumann, Y., Ding, Y.Z. and Rabin, M.O. "Everlasting security in the bounded storage model." *IEEE Trans. Inform. Theory, Special issue on Shannon theory: perspective, trends, and applications*, *48* (6). 12. 2002

[BAL05] Ballani, H. and Francis, P. "Towards a Global IP Anycast Service." *Proceedings of ACM SIGCOMM*. 2005

[BAR04] Barak, B., Canetti, R., Nielsen, J.B. and Pass, R., "Universally Composable Protocols with Relaxed Set-Up Assumptions." in *45th Symposium on Foundations of Computer Science (FOCS 2004)*, (Rome, Italy, 2004), IEEE Computer Society. http://csdl.computer.org/comp/proceedings/focs/2004/2228/00/22280186abs.htm

[BEL06] Belaramani, N., Dahlin, M., Gao, L., Nayate, A., Venkataramani, A., Yalagandula, P. and Zheng, J. "PRACTI Replication." *3rd Symposium on Networked Systems Design and Implementation (NSDI '06)*, San Jose, CA, 2006

[BER03] Berk, V., Bakos, G. and Morris, R. "Designing a framework for active worm detection on global networks." *IEEE International Workshop on Information Assurance*, 2003

[BOR01a] Borisov, N., Goldberg, I. and Wagner, D. "Intercepting Mobile Communications: The Insecurity of 802.11." *Proceedings of MOBICOM 2001*. 2001

[BRE05] Brewer, E., Demmer, M., Du, B., Fall, K., Ho, M., Kam, M., Nedevschi, S., Pal, J., Patra, R. and Surana, S. "The Case for Technology for Developing Regions." *IEEE Computer*, *38* (6). 25–38. 2005

[CAE05] Caesar, M., Caldwell, D., Feamster, N., Rexford, J., Shaikh, A. and van der Merwe, J. "Design and implementation of a routing control platform." *ACM/USENIX NSDI*. 2005

[CAN02] Canetti, R., Lindell, Y., Ostrovsky, R. and Sahai, A. "Universally composable two-party and multi-party secure computation." *Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, ACM, New York, 2002, 494--503 (electronic)

[CAS06] Casado, M., Garfinkel, T., Akella, A., Freedman, M., Boneh, D., McKeown, N. and Shenker, S. "SANE: A Protection Architecture for Enterprise Networks." *Usenix Security*, 2006

[CHE04] Cheung, S.Y., Coleri, S., Dundar, B., Ganesh, S., Tan, C.W. and Varaiya, P. "Traffic Measurement and Vehicle Classification with a Single Magnetic Sensor." *TRB 84th Annual Meeting*, 2004

[CHI] Chiappa, N. "Nimrod Documentation page." http://ana-3.lcs.mit.edu/~jnc/nimrod/docs.html

[CLA03] Clark, D., Sollins, K., Wroclawski, J., Katabi, D., Kulik, J. and Yang, X. "New Arch: Future Generation Internet Architecture (Final Technical Report)." 2003. http://www.isi.edu/newarch/

[CLA03b] Clark, D.D., Partridge, C., Ramming, J.C. and Wroclawski, J.T. "A knowledge plane for the internet." *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '03)*, ACM Press, Karlsruhe, Germany, 2003. http://doi.acm.org/10.1145/863955.8639

[CRA98] Cranor, L.F. and LaMacchia, B.H. "Spam!" *Commun. ACM*, *41* (8). 74-83. 1998

[CSTB03] Committee on Intersections Between Geospatial Information and Information Technology, N.R.C. "IT Roadmap to a Geospatial Future." Computer Science and Telecommunications Board (CSTB), National Research Council of the National Academies, Washington, DC, 2003, 136

[CSTB03b] Committee on the Internet Under Crisis Conditions: Learning from September 11. "The Internet Under Crisis Conditions: Learning from September 11." Computer Science and Telecommunications Board (CSTB), National Research Council of the National Academies, Washington, DC, 2003, 75

[DAM99] Damgård, I.B. "Unconditional Security in Cryptography: Was Shannon too Pessimistic?" *European Association for Theoretical Computer Science*, *68*. 13. 1999

[DEA04] Dean, J. and Ghemawat, S., "MapReduce: Simplified Data Processing on Large Clusters." in *Proc. OSDI 2004*,

[DESIGN] Peterson, L. (ed.), "*GENI Facility Design (to be published)*, 2007

[DEW92] DeWitt, D. and Gray, J. "Parallel database systems: the future of high performance database systems." *Communications of the ACM*, *35* (6). 85-98. 1992

[DOV05] Dovrolis, C., Dhamdhere, A. and Jiang, H., "Buffer Sizing for Congested Internet Links." in *IEEE INFOCOM*, (Miami, FL, 2005)

[DTN04] DARPA. "Delay Tolerant Networking." 2004, Program description and proposal solicitation. http://www.darpa.mil/ato/solicit/DTN

[DWO03] Dwork, C., Goldberg, A. and Naor, M. "On Memory-Bound Functions for Fighting Spam Advances on Cryptology ", *CRYPTO 2003*, Santa Barbara, CA, 2003

[EARTH] Earthscope. http://www.earthscope.org

[ENA06] Enachescu, M., Ganjali, Y., Goel, A., McKeown, N. and Roughgarden, T., "Routers with very small buffers." in *IEEE INFOCOM'06*, (Barcelona, Spain, 2006)

[EST92] Estrin, D., Rekhter, Y. and Hotz, S. "Scalable inter-domain routing architecture." *ACM SIGCOMM Computer Communication Review*, *22* (4). 40-52. 1992

[FAL03] Fall, K. "A delay tolerant network architecture for challenged networks." *Proceedings of ACM SIGCOMM*. 27–31. 2003

[FAL07] Falk, A. "GENI Systems Requirements (to be published)." 2007

[FEL04] Feldmann, A., Maennel, O., Mao, Z.M., Berger, A. and Maggs, B. "Locating internet routing instabilities." *ACM SIGCOMM Computer Communication Review 34* (4). 205-218. 2004

[FOX97] Fox, A., Gribble, S.D., Chawathe, Y., Brewer, E.A. and Gauthier, P. "Cluster-based scalable network services." *Proceedings of the sixteenth ACM symposium on Operating systems principles*. 78-91. 1997

[FRE04] Freedman, M.J., Freudenthal, E. and Mazieres, D., "Democratizing Content Publication with Coral." in *1st Symposium on Networked Systems Design and Implementation (NSDI 2004)*, (2004)

[GAR03a] Garfinkel, T., Pfaff, B., Chow, J., Rosenblum, M. and Boneh, D. "Terra: a virtual machine-based platform for trusted computing." *Proc ACM Symposium on Operating Systems Principles*. 2003

[GAR03b] Garfinkel, T., Rosenblum, M. and Boneh, D., "A Broader Vision for Trusted Computing." in *9th Workshop on Hot Topics in Operating Sys. (HotOS-IX)*, (2003)

[GIB03] Gibbons, P.B., Karp, B., Ke, Y., Nath, S. and Seshan, S. "IrisNet: An Architecture for a World-Wide Sensor Web." *IEEE Pervasive Computing*, *2* (4). 2003

[GOV99] Govindan, R., Alaettinoglu, C., Eddy, G., Kessens, D. and Kumar, S. "An architecture for stable, analyzable Internet routing." *Network, IEEE*, *13* (1). 29-35. 1999

[GRA90] Graefe, G., "Encapsulation of parallelism in the Volcano query processing system." in *Proceedings of the 1990 ACM SIGMOD international conference on Management of data*, (1990), 102-111

[GRE05] Greenberg, A., Hjalmtysson, G., Maltz, D.A., Myers, A., Rexford, J., Xie, G., Yan, H., Zhan, J. and Zhang, H. "A clean slate 4D approach to network control and management." *ACM SIGCOMM Computer Communication Review*, *35* (3). 41-54. 2005

[GYO04] Gyongyi, Z., Garcia-Molina, H. and Pedersen, J. "Combating web spam with TrustRank." *Proceedings of VLDB*, 2004

[HAR02] Harrington, D., Presuhn, R. and Wijnen, B. "RFC 3411: An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks." *Internet Request for Comments*, 2002

[HEL05] Hellerstein, J.M., Paxson, V., Peterson, L., Roscoe, T., Shenker, S. and Wetherall, D. "The Network Oracle." *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, *28* (1). 2005

[HIT06] Ballani, H. and Francis, P., "CONMan: Taking the Complexity out of Network Management." in *Sigcomm Internet Network Management Workshop*, (Pisa, 2006)

[HU04] Hu, Y.C., Perrig, A. and Sirbu, M. "SPV: Secure Path Vector Routing for Securing BGP." *ACM SIGCOMM Computer Communication Review*, *34* (4). 179-192. 2004

[HUL06] Hull, B., Bychkovsky, V., Chen, K., Goraczko, M., Miu, A., Shih, E., Zhang, Y., Balakrishnan, H. and Madden, S. "A Distributed Mobile Sensor Computing System." *SenSys*, Boulder, CO, 2006

[ISH06] Ishai, Y., Kushilevitz, E., Ostrovsky, R. and Sahai, A., "Cryptography from Anonymity." in *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS `06)*, (Berkeley, CA, 2006)

[KER04] Kerravala, Z. "Enterprise Networking and Computing: the Need for Configuration Management." *Yankee Group Report*, 2004

[KHA89] Khan, A.K. and Zinky, J. "The Revised ARPANET Routing Metric." *Proc. fo ACM SIGcomm*, *89*. 45-56. 1989

[LAB00] Labovitz, C., Ahuja, A., Bose, A. and Jahanian, F., "An experimental study of internet routing convergence." in *SIGCOMM 2000*, (2000)

[LEE06] Lee, U., Magistretti, E., Zhou, B., Gerla, M., Bellavista, P. and Corradi, A. "MobEyes: Smart Mobs for Urban Monitoring with a Vehicular Sensor Network." *PerSense Workshop*, (also to appear in IEEE Wireless Communications, 2007), Pisa, Italy, 2006

[LEF02] Le Faucheur, F., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P. and Heinanen, J. "RFC 3270: Multi-Protocol Label Switching (MPLS) Support of Differentiated Services." *Internet Request for Comments*, 2002

[LEN04] Lentczner, M. and Wong, M. "Sender Policy Framework: Authorizing Use of Domains in MAIL FROM." *Internet Draft (work in progress)*, 2004

[MOC87] Mockapetris, P. "RFC 1034: Domain Names – Concepts and Facilities." *Internet Request for Comments*, 1987

[MOR03] Moore, D., Paxson, V., Savage, S., Shannon, C., Staniford, S. and Weaver, N. "Inside the Slammer Worm." *IEEE Security and Privacy*. 2003

[MOS06] Moskowitz, R. and Nikander, P. "RFC 4423: Host Identity Protocol (HIP) Architecture." *Internet Request for Comments*, 2006

[MUR06] Murphy, S. "RFC 4272: BGP Security Vulnerabilities Analysis." *Internet Request for Comments*, 2006

[MUT06] Muthukrishnan, S. *Data Streams: Algorithms and Applications*. Now Publishers, 2006

[NRC04] Committee to Develop a Long-Term Research Agenda for the Network for Earthquake Engineering Simulation (NEES), N.R.C. "Preventing Earthquake Disasters: The Grand Challenge in Earthquake Engineering: A Research Agenda for the Network for Earthquake Engineering Simulation (NEES)." National Research Council of the National Academies, Washington, DC, 2004, 192

[NYB95] Nyberg, C., Barclay, T., Cvetanovic, Z., Gray, J. and Lomet, D. "Alphasort: A cache-sensitive parallel external sort." *The VLDB Journal The International Journal on Very Large Data Bases*, *4* (4). 603-627. 1995

[PAN05] Pan, P., Swallow, G. and Atlas, A. "RFC 4090: Fast Reroute Extensions to RSVP-TE for LSP Tunnels." *Internet Request for Comments*, 2005

[PAR04] Parker, A. "The true picture of peer-to-peer filesharing." 2004. http://www.cachelogic.com

[PAR06] Park, K. and Pai, V.S. "Scale and Performance in the CoBlitz Large-File Distribution Service." *NSDI 06*, 2006

[PAR93] Partridge, C., Mendez, T. and Milliken, W. "RFC 1546: Host Anycasting Service." *Internet Request for Comments*, 1993

[PEI05] Pei, D., Azuma, M., Massey, D. and Zhang, L. "BGP-RCN: improving BGP convergence through root cause notification." *Computer Networks*, *48* (2). 175-194. 2005

[PIE06] Pietzuch, P., Ledlie, J., Shneidman, J., Roussopoulos, M., Welsh, M. and Seltzer, M. "Network-Aware Operator Placement for Stream-Processing Systems." *Proceedings of the 22nd International Conference on Data Engineering (ICDE'06)*, 2006

[QUI02] Qing, X., Hedrick, K., Sengupta, R. and VanderWerf, J., "Effects of vehicle-vehicle/roadside-vehicle communication on adaptive cruise controlled highway systems." in *Proceedings of IEEE Vehicular Technology Conference*, (2002)

[RAM99] Ramsdell, B. "RFC2633: S/MIME Version 3 Message Specification." *Internet Request for Comments*, 1999

[RAT05] Ratnasamy, S., Shenker, S. and McCanne, S. "Towards an evolvable internet architecture." *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*. 313-324. 2005

[REK95] Rekhter, Y. and Li, T. "RFC 1771: A Border Gateway Protocol." *Internet Request for Comments*, 1995

[REX04] Rexford, J., Greenberg, A., Hjalmtysson, G., Maltz, D.A., Myers, A., Xie, G., Zhan, J. and Zhang, H. "Network-Wide Decision Making: Toward A Wafer-Thin Control Plane." *Proceedings of HotNets III*. 2004

[ROS03] Roscoe, T., Hand, S., Isaacs, R., Mortier, R. and Jardetzky, P. "Predicate routing: Enabling controlled networking." *SIGCOMM Comput. Commun. Rev.*, *33* (1). 65-70. 2003

[SES05] Seshadri, A., Luk, M., Shi, E., Perrig, A., Doorn, L.v. and Khosla, P. "Pioneer: verifying code integrity and enforcing untampered code execution on legacy systems." *ACM SIGOPS Operating Systems Review*, *39* (5). 2005

[SHI05] Shieh, A., Williams, D., Sirer, E.G. and Schneider, F.B. "Nexus: a new operating system for trustworthy computing." *Proceedings of the twentieth ACM symposium on Operating systems principles*, 2005

[SIN03] Singh, S., Estan, C., Varghese, G. and Savage, S. "The EarlyBird System for Real-time Detection of Unknown Worms." *HOTNETS-II*, 2003

[SRI01] Srisuresh, P. and Egevang, K. "RFC 3022: Traditional IP network address translator (traditional NAT)." *Internet Request for Comments*, 2001

[STO02] Stoica, I., Adkins, D., Zhuang, S., Shenker, S. and Sonesh Surana "Internet Indirection Infrastructure." *Proceedings of ACM SIGCOMM*. 2002

[TPA] Alliance, T.C.P. "TCPA main specification v. 1.1b. ." http://www.trustedcomputing.org

[VAR00] Varadhan, K., Govindan, R. and Estrin, D. "Persistent route oscillations in inter-domain routing." *Computer Networks*, *32* (1). 1-16. 2000

[WAN04] Wang, L., Park, K., Pang, R., Pai, V.S. and Peterson, L. "Reliability and Security in the CoDeeN Content Distribution Network." *USENIX '04*, 2004

[WIKI06] "Communication during the September 11, 2001 attacks." *Wikipedia, The Free Encyclopedia*, 2006. http://en.wikipedia.org/wiki/Communication_during_the_September_11,_2001_attacks

[WIS05] Wischik, D. and McKeown, N. "Buffer sizes for core routers." *ACM/SIGCOMM CCR*, 2005

[YAN02] Group, Y. "The Yankee Group 2002 Network Downtime Survey." 2002

[YAN04] Yang, X. "NIRA: A New Internet Routing Architecture." *PhD Thesis, Computer Science and Artificial Intelligence Laboratory*, Massachusetts Institute of Technology, Cambridge, MA, 2004. http://hdl.handle.net/1721.1/30495

[YEH04] Yehuda, L. "Lower bounds for concurrent self composition." in *Theory of cryptography*, Springer, Berlin, 2004, 203--222

[ZHA05] Zhang-Shen, R. and McKeown, N. "Designing a Predictable Internet Backbone with Valiant Load-Balancing." *Thirteenth International Workshop on Quality of Service (IWQoS 2005)*, Passau, Germany, 2005

[ZHO05] Zhou, P., Nadeem, T., Kang, P., Borcea, C. and Iftode, L. "EZCab: A Cab Booking Application Using Short-Range Wireless Communication." *Proceedings of Third IEEE International Conference on Pervasive Computing and Communications (PerCom 2005)*, 2005, 27-38

7 Appendix: Non-Research Issues

While GENI's potential for transforming research is the focus of this document, there are several other issues that must be addressed in the MREFC process. At this early stage in the process, and with the GENI project caught midway between the old GENI Planning Group and the newly created GENI Science Council (GSC) and soon-to-be-selected GENI Project Office (GPO), the plans for addressing these other issues are very preliminary. Below we briefly sketch our current views on these areas, but more concrete plans will have to await the firm establishment of the GPO and GSC.

7.1 Education

Currently most networking students learn mainly from a textbook, with perhaps some projects to give them implementation experience. Nowhere are they able to get hands-on experience with large-scale deployments. This impacts both the training of our future workforce and the quality of networking research.

GENI can change this. Imagine being able to demonstrate various alternative designs (of, say, routing protocols) to a class not just on a powerpoint slide, or in simulation, but actually deployed on hundreds of GENI nodes with real-time performance measurements and online diagnosis of any problems that occur. By providing this opportunity, GENI promises to be a powerful educational tool.

Moreover, GENI will encourage the sharing of educational material. The networking and distributed systems communities have considerable past experience in developing and using shared infrastructures for educational purposes. Recent efforts include lab and courseware kits, programmable wired and wireless networks, emulation environments, and other experimental platforms such as PlanetLab, Orbit, and Emulab. We expect to draw on best practices and lessons learned from these educational efforts as we plan for GENI.

7.2 Outreach

One can view GENI as "democratizing" networking research, in that it makes a powerful experimental facility available to anyone with an Internet connection. This "lowers the barriers to entry" and will broaden the community of students who can engage in cutting-edge research.

At this early stage there are no plans in place for specific outreach activities to particular under-represented populations. These we expect will be put in place somewhat later in the MREFC process. However, it should be clear that GENI provides a powerful and accessible foundation upon which future outreach activities can be based.

7.3 International Cooperation

The GENI effort has already been in contact with many other international efforts. For example, joint EU-NSF workshops have been organized to focus on both the research agenda and the requirements for experimental facilities. No international cooperation plans have been formalized because GENI is not far enough along in the MREFC process, but the interest around the world is obvious. Moreover, GENI is expressly designed to support federation of facilities, thereby enabling other countries and organizations to offer network resources to a broader experimental facility; these network resources will still be controlled by their own local policy but can be used by a broader community of researchers.

7.4 Industrial Participation

Industrial participation in GENI will be mostly coordinated by the GPO. Given that the GPO has not yet been announced (at the time of this writing), no detailed plans for industrial involvement have been made. However, there is an immense opportunity for industrial participation, ranging from providing specific technologies (e.g., routers, optical devices), to providing links, to providing management expertise, to providing a wide variety of edge devices (e.g., phones, sensors). These opportunities will be more fully explored in the coming months as the GPO begins operation.