Conceptual Design
Project Execution Plan

GDD-06-07

GENI: Global Environment for Network Innovations

January 10, 2006

The following planning group is advising NSF on the design of the GENI facility and has prepared this Conceptual Design and Project Execution Plan document.

Tom Anderson, *University of Washington*

Dan Blumenthal, *University of California Santa Barbara*

Dean Casey, *NGENET Research*

David Clark, *Massachusetts Institute of Technology*

Deborah Estrin, *University of California Los Angeles*

Larry Peterson, *Princeton University* (Chair)

Dipankar Raychaudhuri, *Rutgers University*

Jennifer Rexford, *Princeton University*

Scott Shenker, *University of California Berkeley*

John Wroclawski, *Information Sciences Institute*

The document defines GENI at a certain level of specificity, but the planning group understands that the underlying technology will change rapidly, and that the requirements the community places on GENI will continue to mature. Therefore, this document should be viewed as a snapshot of GENI as of January 10, 2006. Additional snapshots will be posted as they come into focus, based on community feedback.

Table of Contents

Executive Summary

This document describes GENI—a Global Environment for Network Innovations—an experimental facility intended to enable fundamental innovations in networking and distributed systems. The essence of the GENI facility is its ability to rapidly and effectively embed *within* itself a broad range of experimental networks, interconnect these experimental networks as appropriate with other experiments and the existing Internet, provide these networks with users and an operating environment, and rigorously observe, measure, and record the resulting experimental outcomes. The proposed facility will span a range of extant and emerging technologies (e.g., wireless sensors, mobile wireless, high function optical), layers of network architecture (e.g., physical to network to network services), geographic reach (e.g., wearable personal area networks to wide area inter-continental networks), and application domains (e.g., high bandwidth, compute intensive e-science to low bandwidth, low duty-cycle sensor applications to large scale information dissemination).

GENI is a unique facility. Unlike traditional network testbeds that demonstrate a single design point, GENI is a general-purpose facility that places essentially no limits on the network architectures, services, and applications that can be evaluated. Unlike traditional network testbeds that either limit researchers to incremental changes or limit researchers to synthetic workloads, GENI is designed to allow both clean-slate designs *and* experimentation with real users under real-world conditions. Unlike traditional testbeds that provide no credible deployment path to the commercial world, GENI represents a model in which incremental adoption of new services has the potential to drive wide-spread deployment.

GENI provides these capabilities through an innovative combination of techniques: *virtualization*, *programmability, controlled interconnection,* and *modularity*. Virtualizing the physical hardware allows multiple network architectures and services to run simultaneously. This includes long-running services and applications that attract real users, which results in realistic evaluations and drives adoption and deployment. Programmability allows clean-slate designs to run side-by-side with incremental experiments. Controlled interconnection allows a wide variety of experiments to build on and interoperate with each other, while providing for each experiment an appropriate controlled-risk environment. GENI's modular design structure explicitly accommodates the evolution of new building block technologies and components over time. This same structure supports federation, which allows other countries and research communities to "plug into" GENI.

GENI will allow its constituent research communities to address fundamental policy and engineering trade-offs in the design of secured, privacy protecting, robust networked systems; self-evolving networks with billions of wireless devices; new or hybrid paradigms of communication suited for high-speed optics that build on today's packet and circuit switching; new models of information dissemination and sharing; co-design of data, new perspectives for structuring the control, and management planes of a network; and others. The shared objective of GENI's many target communities is to create a Future Internet that reliably meets the requirements of today and is fully prepared for the challenges of tomorrow. GENI's capabilities provide a crucial, and otherwise unobtainable, stepping stone on the path towards this objective.

Motivating Research Challenges

Today's Internet, based on design decisions made in the 1970's, is extraordinarily successful. It is remarkable that only now, after 30 years, assumptions built into its design begin to limit its potential. These design assumptions cannot be removed by minor incremental adjustments 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. In particular, it should meet the needs of society in times of crisis by giving priority to critical communications.
- 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. A large number of specific problems with the Internet today have their roots in an economic disincentive, rather than a technical lack.
- 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 the Internet in the home have turned many users away, limiting the future penetration of the Internet into society.

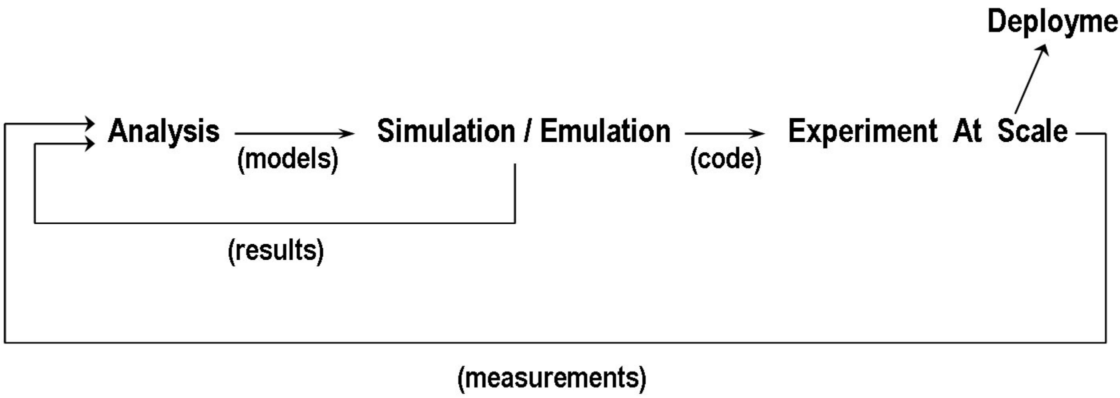
These limitations are deeply rooted in the design of the Internet. It is easy to overlook them because of the astonishing success of the Internet to this point. In the mere decade since the Internet left the research arena and entered the commercial world, it has substantially changed the way we work, 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.*

A second, equally important motivation complements the imperative for action to correct these limitations. This second motivation is that the Future Internet foster, rather than inhibit, emerging applications and technologies. Imagining that the Internet simply does better what it already does today is a very narrow view of the future. Yet, for a variety of reasons, the Internet today is poorly positioned to accommodate this blossoming of new capabilities. To realize its potential, a Future Internet must enable and encourage:

- 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.

The Need for GENI

A key element of any effort to redesign the Internet is a strategy for fostering the research *cycle*—drastically lowering the barriers that promising new directions developed by the research community face before transition to industrial development and deployment within the commercial Internet. This requires that we move well beyond the methodologies and facilities used today. An experimental facility that enables the research community to address the questions outlined in earlier sections is a key part of a seamless, end-to-end research process for taking ideas from conception, through validation, to deployment, similar to the idealized process shown below.



Unfortunately, it is well known within the networking research community that we lack effective methodologies and tools for rigorously evaluating, testing, and deploying new ideas. Today, most new ideas for how the Internet might be improved are initially evaluated using simulation, emulation, and theoretical analysis. These techniques are invaluable in helping researchers understand how an algorithm or protocol operates in a controlled environment, but they provide only a first step. The problem is that the simulations tend to be based on models that are backed by conjecture rather than empirical data; models that are overly simple in virtually every attribute of practical significance—topologies, administrative policies, workloads, device failures, and so on. Theoretical analysis, while immensely valuable in answering certain types of questions, remains today even more limited in the realism of the problems that can be addressed. Thus, true understanding of complex protocols and comprehensive network architectures demands extensive live experimentation. Toward this end, testbeds are the conventional second step in the research process.

However, current testbed paradigms are inadequate to this task, largely due to what the community refers to as the *testbed dilemma*. Traditional testbeds can be roughly categorized as production-oriented or research-oriented [AND05]. Production testbeds, such as Internet2 [I2], support real traffic from real users, often in large volume and across many sites. As such, they provide valuable information about the operational behavior of network architectures, protocols, and subsystems, but because users depend on them, production testbeds must be extremely conservative in their experimentation, using well-honed implementations of incremental changes. In contrast, research testbeds (such as DETER [DET]) do not carry traffic from a wide variety of real users but instead are typically driven by synthetically generated traffic and/or a small collection of explicitly willing users. This allows them to be much more adventurous, capable of running first-cut implementations of radically new designs. Unfortunately, this lack of real traffic also renders the results much less indicative of real operational viability. As a result, neither kind of testbed—production or research—produces the data needed to adequately evaluate new architectures. It is therefore difficult to make a compelling case for new architectural designs based on a classical testbed evaluation.

A second limitation of traditional testbeds is that the community must know what testbed it wants to build. If the GENI facility's motivating research was narrowly conceived as resulting from exactly one proposal for one new Internet architecture, and the subsequent demonstration of that proposal, then a logical approach to experimental infrastructure would be a purpose-built testbed targeted to that single proposal. However, this view does not reflect the reality of how the research GENI supports will proceed. In seeking answers to the many research challenges outlined in this document, the best ideas will emerge from a competition among different intellectual perspectives and concrete proposals. Ideas may merge and diverge, and different options will require testing and validation. At any moment, there will be different ideas with different maturity and reflecting different approaches. This means that an experimental infrastructure must support multiple, simultaneous, logically different, long-lived experiments, trials and demonstrations. At the same time, it must support the researcher with a suite of tools, libraries, and components to facilitate and support the core science, so that individual researchers do not need to reinvent every wheel. Taken together, these are the defining objectives of the GENI design.

The GENI Facility

The GENI design is based on several key concepts. At the lowest level, GENI comprises a collection of substrate hardware resources, including nodes, links and edge subnets. Each experiment using GENI will run on some subset of the GENI resources. 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 substrate will include management software that is used to allocate resources to slices, and ensure that slices do not interfere with each other. While GENI will initially incorporate a narrow range of resources and simple assignment policies, the plan is for this range to advance over time.

The GENI design supports 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.

To support this vision, the basic design of GENI is divided into two parts: (1) a *physical network substrate*, and (2) a *global management framework*. We describe each briefly.

Physical Network Substrate

The physical network substrate consists of an expandable collection of *building block* components. Although no single building block could do so by itself, the set of building blocks 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, the GENI execution plan defines an initial set of building blocks to be deployed:

- **Flexible edge devices** intended to provide the computational resources needed to build wide-area services and applications, as well as initial implementation of new network elements.
- **Customizable High-Speed Routers** intended to implement core network data processing functions for high-speed, high volume traffic flows.
- **Dynamic Optical Switching Components** intended to provide data handling in the optical domain at the circuit, burst, or packet level, and with increasing functionality as optical technology further develops.
- A **National Fiber Facility** intended to provide 10Gbps or higher light path interconnection between GENI core nodes.
- A large number of **tail circuits** of varying technology, intended to connect GENI edge sites to the GENI core, and the GENI core to the current 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 **congnitive 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 Subnets** that support controlled experiments by allowing researchers to introduce and utilize artificially generated traffic and network conditions within an experimental framework.

Global Management Framework

The second major part of GENI, the global management framework, knits the building blocks 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 overlaying slices onto the GENI substrate, and controlling these slices on behalf of experimenters.

Perhaps the most 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 independently of each other. The 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. Similarly, outside organizations that contribute their own resources can federate with GENI, while retaining autonomous control over their components.

This decentralized design approach provides several important capabilities; among these are GENI’s ability to evolve over time to accommodate new technologies and research objectives, as well as the ability for different developers and research communities to contribute infrastructure to GENI within a loosely coordinated project management model.

Design Capabilities

Taken together, GENI’s decentralized, slice-based design elements allow it to meet a number of objectives. Among the most important of these are:

- **Service/architecture neutrality:** What is most important for research in network architecture and services is that the level of abstraction be low enough to permit full experimentation. Different slices of the GENI may support different experiments at the same time
- **Edge diversity:** GENI enables heterogeneity in the end systems that connect to it and participate in the experiments running within it. In particular, it enables the connection of limited functionality end-systems (such as wireless PDAs and sensors) connected by a variety of technologies (such as wireless and sensor networks).
- **Ease of user access:** Mechanisms are provided to make it easy for users to join one or more experimental services running in GENI, and to transparently fall back to the legacy Internet

whenever the experimental network cannot provide the requested service. This capability supports the requirement that GENI experiments attract and retain real users.

- **Global reach:** To support experimentation at scale, and to maximize the opportunity to attract real users, GENI must have as wide of reach as possible. Access cannot be limited to only those few sites that host backbone nodes.
- **Instrumentation and data analysis:** The GENI substrate, along with all the architectures and services deployed on it, will be heavily instrumented, and generated data collected and archived, and analysis tools developed, through the use of independently evolving services incorporated within the overall design framework.
- **Federation and sustainability:** To ensure the sustainability of GENI, it will be possible for participating institutions to contribute resources in return for access to the resources of the GENI as a whole. Further, it will be possible for other research communities to ``opt-in'' by connecting purpose-built networks (including dedicated transmission pipes and sensor networks) into the GENI substrate and running their applications in a slice of GENI.
- **Inter-slice composition:** GENI will enable interconnection among slices by mutual consent, and between slices and the legacy Internet. This permits slices to host network services with external users, and/or to act as transit networks. Nothing will prevent a researcher from inter-connecting a virtual network running within a slice with another network. This other network could be running within another slice of GENI, or it could be the legacy Internet or another custom network (or testbed) that runs over standard IP protocols.
- **Policy and governance:** Because GENI comprises shared infrastructure, the technical design must support a governance process to guide allocation of resources to slices. GENI's architecture allows project management to that implement and enforces such policies.

Project Management and Construction

The GENI project will be hosted by a research consortium that serves as prime contractor and ultimate management authority. The **GENI Community Consortium** (GCC) will be a member-based organization in which scientific, educational, and research institutions will be eligible to apply for membership. The GCC will be a 501(c)(3) non-profit corporation, and as such, will have a Board of Directors with fiduciary responsibility. The GCC will have additional organizational structure, to be elaborated at the time it is created. Here, we focus on those aspects of the GCC most relevant to GENI.

The GCC will establish a standing **Executive Committee** (EC) to oversee the GENI project. The EC will consist of senior members of the academic, corporate, and government research communities, with the restriction that EC members will not compete for sub-contracts to build GENI.

The EC will have four major responsibilities. The first is to appoint a Project Director that is ultimately responsible for the project as a whole, and a Project Manager that is responsible for the management and execution of the project. The Director is answerable to the EC and the Manager is a full-time employee of the GCC and reports to the Project Director. The second duty of the EC is to create a **Technical Advisory Board** (TAB) that provides technical leadership for the project, largely be creating a set of working groups that consider narrowly-defined technical issues. The TAB is chaired by a senior scientist that serves as Chief Architect for GENI; the Project Director serves as an ex-officio member of the TAB. The third duty of the EC is to create a **Project Management Office** (PMO), to be directed by the Project Manager. The PMO includes financial, operations and planning, workflow, and system engineering office, among others. The forth duty of the EC is to conduct a fair and open competition through which the development teams responsible for building GENI are selected. This will involve soliciting proposals, running a set of review panels to evaluate the proposals, and finally selecting the teams to be awarded sub-contracts. (The awards will actually be executed by the PMO, which will also monitor the performance of the awardees and adjust sub-contracts as conditions warrant.)

We break down the work required to build GENI into four major tasks, each of which is further divided into 2-5 additional sub-tasks. The first three major tasks involve significant development efforts. The fourth major task involves assembling the building block components into a single comprehensive infrastructure, plus on-going management of that infrastructure. These tasks are outlined below:

- **Node Development:** Work is required to realize the several types of node technologies to be included in GENI, with each to be completed at different times during the five-year schedule. For each, the development task involves a combination of assembling and testing the base hardware components, and writing the component manager and control protocols that each node needs to support in order to “plug into” the GENI framework.

- **Wireless Subnet Development:** Work is required to build the five types of wireless subnets we expect to connect to GENI. For each, the development task involves selecting appropriate sites, installing access points, distributing assorted edge devices, and writing the component manager and control software that each subnet needs to support in order to “plug into” the GENI framework.

- **Management Software Development:** Work is required to develop the various software management modules and services. These include the GENI Management Core, necessary infrastructure services, a collection of underlay services, and the glue modules that allow external systems to interact with GENI. The software architecture is defined in such a way that each of these software systems can be developed independent of each other. Moreover, we expect each software system to continually evolve over the course of the project.

- **Network Assembly and Management:** Work is required to connect the component node and subnet technologies into an end-to-end facility. This involves acquiring the necessary fiber to build a national backbone, populating each PoP of this backbone with the appropriate node types, installing PC clusters and wireless subnets at appropriate edge sites, connecting these edge sites to backbone PoPs using the most appropriate tail circuits, and interconnecting a subset of the PoPs to the legacy Internet via the appropriate Internet Exchanges. It also involves on-going management of the resulting network.

The work breakdown assigns one or more teams to each sub-task. For the development sub-tasks, these teams include a lead architect and an appropriate mix of software engineers and hardware technicians. For those development sub-tasks that have been assigned multiple teams, we expect the teams to operate independent of each other; they are primarily pursuing parallel or independent aspects of the corresponding task. The annualized five-year budget is summarized as follows:

TOTAL	2009	2010	2011	2012	2013
$367,411k	$85,460k	$69,655k	$66,372k	$74,342k	$71,582k

Conclusion

It is both critical and interesting to ask how GENI might meet its larger goals. We suggest that success may come in multiple flavors. By demonstrating the value of a new architecture, particularly for demanding applications, we may facilitate the modification of the existing Internet to incorporate a set of new ideas and functionality. Or, it may be that by developing a set of advanced services well beyond what the current Internet can easily provide, users come to rely on these services, rendering the underlying Internet less and less important or visible to end users. Or it may be that a ``next generation'' architecture, after having been validated on GENI, would, through some magical process of consensus and daring, be adopted by ISPs and router vendors alike.

To us, an *organic* deployment story seems most likely. In this organic story, there is no discrete or global decision point at which the old world accepts and incorporates the new technology; the process is continuous and incremental by definition. The players that represent the old order may respond to market opportunities, for example, by providing high-performance or more cost-effective implementations of the new technology demonstrated on GENI. Simultaneously, the uses that rely on the unique characteristics of GENI—its security, reliability, and flexibility with respect to new application domains—could over time become more and more prevalent, so that increasing numbers of users have their Internet use mediated by GENI itself, or by services originally launched on GENI.

As improbable as this organic story may sound, there is at least one existence proof that it works: the Internet itself. Both the original ARPANET and the Internet that followed began as overlays running on top of the entrenched telephony system. The disruptive Internet technology eventually transformed the underlying telephony system from being circuit-based to being packet-based. Today, it is difficult to say where the old technology ends and the new technology begins.

1 Introduction

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. In particular, it should meet the needs of society in times of crisis by giving priority to critical communications.
- 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 Internet-based telephone service. Similarly, mismatches between local-area networks and long-haul transport economics create barriers to end-to-end Gbps to the desktop. A large number of specific problems with the Internet today have their roots in an economic disincentive, rather than a technical lack.
- 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. That is the motivation for this effort.

This document describes the experimental research facility needed to address these challenges—to correct the limitations in the current design and to explore the opportunities to make the Internet an even more valuable tool. The proposed facility—called GENI (Global Environment for Network Innovations)—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 new 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. In sum, GENI will support a seamless research process for taking ideas from conception, through validation, to deployment.

The remainder of this document considers first, the scientific goals to be enabled by GENI—to answer how we would design a better Internet. Second, we discuss why these scientific questions cannot be answered without a large-scale infrastructure. We conclude with a detailed description of the proposed infrastructure: its components, construction, and management.

2 Research Goals

The research challenge at the center of this document is to understand how to design an Internet that achieves its potential. The Internet has been a fantastic success, but in many ways it is not meeting the needs of its users. Today, it is not secure, hard to use, and unpredictable. Its technical design has created barriers rather than stimulants to key industrial investments. Tomorrow, it needs to support emerging computing technologies, new network technologies such as wireless, and emerging applications. Getting from where we are now to a new concept for an Internet is a goal of critical national importance.

We characterize the research agenda along two orthogonal axes. The first primarily focuses on issues of design—creating, evaluating, and synthesizing the mechanisms and architectural features that realize the Future Internet. Section 2.1 presents the research agenda from a design perspective. The second axis considers cross-cutting foundational questions—modeling, analyzing, and formalizing the limits and properties of the Future Internet. Section 2.2 presents the research agenda from a foundational perspective. Note that we use these two axes primarily for purposes of presentation; individual researchers typically pursue design and foundational questions simultaneously.

2.1 Design Challenges and Opportunities

This section summarizes the important requirements and opportunities for the design of a Future Internet. 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. The 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.

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 mobility 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 give 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 just 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 [BOR01]. 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.

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 device 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.

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.

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 Service in Times of Crisis

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 much less to social needs. A very important example of a collective social need is service in times of crisis. For most consumers, of course, their access to the Internet is not even designed to stay up when the power goes down, so a disaster renders the Internet useless today. On the other hand, 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 strategies 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.

Much of the work on supporting citizens in times of crisis is done within the social sciences. This is another opportunity to reach out to other parts of NSF as a part of this project.

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.

2.1.6 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.

2.1.7 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 early designers of the Internet may not have fully understood this, but 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.) Any redesign of the Internet needs to consider how to encourage 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 ones 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 define 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.

2.2 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.

2.2.1 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.

2.2.2 Architectural Limits

A fundamental question at the core of this effort is to understand the architectural limits of the current Internet, and to test whether alternative designs better position the Internet to address the many challenges it faces. At the heart of this question is the issue of whether or not we can continue to patch the Internet for the indefinite future, or are there indeed limits to the current design that will keep the Future Internet from realizing its potential.

While there is no way to be certain that the incremental path we are currently following will ultimately fail to address the challenges facing the Internet, it is clear that many of the assumptions underlying the Internet's design no longer hold:

- The Internet originally viewed network traffic as fundamentally friendly, but it today it is more appropriate to view it as adversarial. An alternative design would minimize trust assumptions.

- **T**he Internet was originally developed independent of any commercial considerations, but today the network architecture must take competition and economic incentives into account. An alternative design would enable more user choice.
- **T**he Internet originally assumed host computers were connected to the edges of the network, but host-centric assumptions are not appropriate in a world with an increasing number of sensors and mobile devices. An alternative design would allow for much more edge diversity.
- **T**he Internet originally did not expose information about its internal configuration, but there is value to both users and network administrators in making the network more transparent. An alternative design would provide more network transparency.
- **T**he Internet originally provided only a best-effort packet delivery service, but there is value in enhancing (adding functionality to) the network to meet application requirements. An alternative design would provide more explicit support for widely distributed applications.
- The Internet originally drew a sharp line between the network and the underlying transport facilities, but emerging optical integration technology makes it possible to embed network functionality in the optical transport. An alternative design would make configurable aspects of the underlying transport a first-class element in the architecture.

Three additional points are worth making. First, it may be possible to solve some of these limitations with incremental point-solutions; however, doing so comes at the cost of increased complexity, which makes it hard to reason about the network as a whole. This increased complexity makes the Internet harder to manage, more brittle in the face of new requirements, and more vulnerable to emerging threats. Understanding the tradeoffs between complexity and architectural purity will be important.

Second, it is possible to overlay new network architectures and services on top of the current Internet without changing the Internet architecture, per se. This assumes the new architecture or service has many points-of-presence, which is a capability that GENI will provide. Understanding the limits of overlay-based solutions, along with identifying what changes to the core network (if any) are necessary to better support overlays, will be a central question addressed by this effort.

Third, 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.

2.2.3 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 of where 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 citizen 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 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.

2.2.4 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).

Wireless is perhaps the most transforming of the current network technologies, with its promise of "always connected", 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, point-to-point circuits, where circuits through ring and mesh networks are actually configured 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.

2.2.5 Broader Interdisciplinary Implications

Beyond looking across boundaries that separate technical sub-communities, this effort will benefit greatly from looking for help from disciplines much farther afield, disciplines as diverse as economics, sociology, and law. 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 being holding users responsible for their actions. Several engineering design points are possible, but it is a legal and societal question as to how this question is resolved. Similarly, there are countless economic issues involved in who extracts value from the network, how cost recovery is managed, and how the network provides incentives for desired behavior.

3 Current Research Landscape

While the case to reconsider the design of the Internet is compelling, it does not necessarily tell us how to best undertake such an endeavor. Nor does it identify who is best positioned to make a difference. This section considers these questions, and in the process, lays out the current research landscape. The discussion effectively summarizes a dialogue currently taking place in the research community, as reported in recent NSF workshop reports [AND05, BLU05, LIS05, PER05, RAY05b]. The main take away from this discussion is that the research community will play a key role in the Internet’s redesign, leveraging a large body of groundwork research done over the last few years.

The Internet’s original design arose from research investments by NSF and DARPA. This investment catalyzed a broad and cooperative research program, and eventually a nascent new industry. The Internet is now fully commercial, so one might assume that a purely private sector activity would lead the next design effort. However, a number of factors argue against this conclusion. Prime among these is the requirement for a careful and systematic in-depth review of the Internet’s architecture, rather than the continuing series of incremental adjustments that characterize a mature industry. Second is the need for a coherent and collaborative cross-community effort to foster the transition—an activity of clear benefit to society and all of its participants, including the networking industry, but one that is most likely to succeed if structured within the pre-competitive environment of the academic research enterprise.

3.1 Why Industry Alone Will Not Solve the Problem

Commercially successful sectors such as Information and Communications Technology have private sector industrial research laboratories, so one might reasonably ask why the private sector will not take on the job of meeting the requirements discussed above. In fact, we cannot reasonably expect them to do so, and this is a justification for the proposed effort.

First, as has been noted many times [CSTB99], there is a qualitative difference between research done in the private sector and research done by academia using government funding.

Companies are motivated to fund research when they can own the results of the research. When the results can also benefit their competitors, it is hard to justify corporate investment. This means that work that leads to a new over-arching design, which equally benefits all players, is hard to justify in the private sector.

Second, there is a problem of leadership. A large player in the market might be able to justify investment in a re-thinking that shifts the whole marketplace, but they would not be trusted to take on this role, exactly because of the potential for abuse of market power. Imagine the reaction if Microsoft were to announce that it was going to redesign the Internet. So while a small company will simply lack the resources that it could justify for this sort of activity, a large company may be held back from offering to take a leadership role.

Third, the private sector has different motivations to shape the future of the Internet. When proposing to make a large research commitment, it is important to ask: "what will happen if we don't do this?" In the case of scientific endeavor, the answer is that we slow the pace of discovery, discourage the people in the profession, and harm our educational and research institutions. The priorities among different scientific projects are usually resolved within that context. In the case of an engineering innovation such as the Internet, the answer is different, because it sits in a larger context of players. If NSF does not undertake this work, the Internet will not sit still. It will evolve under the priorities of the private sector players who are investing in it today. Based on what we can already see, this will result in a narrowing of the goals of the Internet, and a loss of utility to broad sectors of Internet users. We will see specific solutions to some of the problems above, point solutions that preclude much of what we might value in an Internet of tomorrow.

For example, it has been proposed that to secure the Internet, no users should be allowed to connect unless they provide a credit card number that is placed in escrow to be used to trace them if they misbehave. The fact that this would exclude the bottom economic tiers of our society and much of the developing world is not material if the view is that the Internet should become a vehicle for e-commerce. (This is not a hypothetical example, but a real proposal.) Another example of "narrowing" is the gradual shift, easily perceptible today, from the Internet as an open platform suited for third-party innovation to a more closed platform where the Internet Service Providers control what applications users can run. This shift is easily understood—it is a natural response to the industry structure induced by the existing Internet design. As we discuss above, ISPs are not in a position to re-conceive that industry structure; they are a part of it. However, if we can propose a slightly different economic model and shift the industry toward it, we may be able to rebalance this tendency and reverse this shift.

The fact of the Internet—what it is and how it works—is a result of its early roots as a product of government-funded (NSF and DARPA) research. It is hard to imagine that private sector investment would have brought us the Internet we have today. But the drive toward tomorrow is now being led by the private sector. It is not too late for the public sector to turn its attention back to the future of the Internet, but it is certainly not too soon. NSF and its research community are clear that now is the time to attend to the future of the Internet, and take a position on what we want it to be in 10 years.

3.2 Why “Research as Usual” Will Not Solve the Problem

The success of the Internet is actually a barrier to radical thinking. An individual CS researcher, considering what problem to address, takes a great risk with an innovation that is not compatible with the Internet of today. How would such an idea be tested? How would it influence the world if it cannot fit into the Internet? Researchers working on the next generation of physical layer technology face barriers related to the risk of disturbing the existing stable infrastructure. How will these new technologies be tested on a network scale? How will prototype control and management software be validated under realistic workloads? For such researchers, the safer and more productive lines of research are those that are incremental improvements to the Internet as it is now. There is little chance of success for one person taking a bold leap into the future, but a bold leap is what is required if we are to envision a radically different future.

This NSF initiative is based on the premise that to achieve a substantive change in what the Internet might be, the research community must accept a challenge—not to ask how we can make the Internet a little better through a small change, but instead to envision an end point—what Internet we want in 10 years, and how we would design it. The process of incremental change, without an overall vision for where we are going, will not move us to any specific long-term objective. As Yogi Berra said, "You've got to be very careful if you don't know where you're going, because you might not get there."

In order to envision a Future Internet that might be rather different from that of today, one must avoid the trap of taking the present network as a given. Our approach is framed around the question of how we would design an internetwork if we could do it “from scratch” today, knowing what we now know about requirements and mechanisms, learning from the past, taking what is good, proposing new approaches where they are needed, and fitting these ideas into a fresh overall architecture. In this respect, the process of design has been called “clean slate” in that the research community is encouraged not to be constrained by features of the existing network. The challenge is not change for the sake of change, but to make an informed speculation about future-looking requirements, to reason carefully about architectural responses to these requirements, to stimulate creative research in innovative and original networking concepts to meet these requirements, and to produce a sound and defensible argument in support of the architecture(s) proposed.

However, this challenge and this approach requires that research be structured in a very different way. First, this goal requires the collective buy-in from a large segment of the community. No single person can succeed in a venture of this scope. Second, members of the community must be motivated to work together toward a common goal—the re-conception of the Internet and the convergence toward an integrated view of its architecture. Finally, there must be some way to try out this new design—some way to test concepts and prove them with real applications and real users. Without some way to validate new ideas, there is little motivation to propose them. This last requirement is the justification for this project.

How might the success of this project be defined? One answer might be that a new architecture for an Internet is developed and then deployed wholesale in place of the current Internet. There are some who believe that this is the only way to shift us to a materially different place—the current Internet, which was initially architected in the 1970's and has been evolving incrementally since then, may now carry enough baggage from its three decades of evolution that it cannot continue to evolve to meet the requirements which we expect it will face in the next decade. Another reasonable and successful outcome for this research would be that by setting a long-term vision for where the Internet should go, the research would help shape and inspire more creative evolution of the existing Internet toward that goal. If the research community can set a vision of what the Internet of ten years should be, and set us on a path to get there, we will have succeeded.

3.3 Laying the Groundwork

Over the last several years, there has been an increasing recognition among the networking research community of the importance of addressing the fundamental limitations of the Internet. The result is a wealth of research that lays the groundwork for this architectural effort. Many aspects of the original Internet design have been reconsidered:

- Addressing: there are many proposed alternatives to the original global addressing mode of the Internet, including private interconnected address spaces with address rewriting [FRA01, NG01], address indirection [I3], and service-level addressing [WAN02]. There are proposals for addressing to support mobility [PER02] and protection from attack [FER98].
- Routing: much research is currently focused on improvements and alternatives to the current inter-provider routing scheme, BGP [CAE05, REX04]. There are also proposals for user-directed routing [CLA89, YAN03, SNO04], gradient routing [POO00], resilient routing [AND01], and dispersion routing [GUS97].
- Capacity management and congestion control: There is more than a decade of research on alternatives to the current Internet congestion control scheme and support for explicit Quality of Service [BRA94, SUB02].
- Rich delivery services: there are a number of proposals for alternatives or additions to the basic delivery service of the Internet, ranging from anycast [PAR93] to multicast [DEE89, CHU00, QUI01], diffusion and data-driven routing [IGE00, GRI01].
- Information-centric architecture: there have been a number of proposals for information naming, search and finding, including proposals for Universal Resource Identifiers / Names [BER94], the CNRI Handle System [KAH95], and other approaches to persistent, layered naming [BAL04].
- Security services: there is a wealth of proposals, including identification and filtering of hostile traffic [SNO02], detection and quarantine of Denial of Service attacks [IOA02], and architectural modifications that might lead to a more secure system, including different addressing modes, routing schemes and replication and randomization of resources [AND03].
- Packets and layering: proposals include Role Based Architecture [BRA02] as an alternative to strict layering, and Application Level Framing [CLA90]. There are proposals to enhance the semantics and performance of the Internet by allowing applications to have selective visibility and control over network-specific technology features [DEC00, KAR02].

- Measurement and monitoring: there is a wealth of work on measuring and characterizing the Internet [SPR02, SPR03]. Much of this has to be done indirectly today, because the current architecture does not provide the best hooks for instrumentation.
- Enhanced network services: new functionality is being deployed throughout the network; examples include file sharing and network-embedded storage [DAB01, KUB00, ROW01a], content distribution networks [FRE04, WAN02], and scalable object location services [BAL02, RAT02, ROW01b, STO01].
- Wireless / mobile networks: algorithms and protocols to support dynamic mobility in heterogeneous networks[JOH04, BAN03]; integrating location services [KUB01, HEL03, LI00]; ad hoc network self-organization and routing [LAN03, GAN04, COU03, KAR00]; security and privacy, decentralized trust [GRU03, BER03, KON02]; cross-layer algorithms [PAU03, TOUM03] and cognitive radio systems [MIT99, MIN05, ACK04].
- Sensor networks [CUL04]: integrating lightweight sensor protocols [HEI99, EST99, HEI01]; dynamic discovery methods [GAN04, INT00]; in-network programming models [SRI00, RAT02b]; content awareness and data integrity / aggregation [GRU03, BOH03,INT02, PRZ03, PERR02]; low latency interaction[SOU04]; socket layer abstractions [SZE00, LEV02].

What all these topics have in common is that most of them have not made their way into the production Internet. While the Internet was designed to make use of new technology, incorporating new architectural alternatives has proved very hard. But these ideas can form the starting point for discussion and development of a Future Internet.

3.4 Strategic Choices

While we have presented the over arching research agenda as one of “reinventing the Internet” and emphasized the importance of allowing a “clean slate design”, we recognize that researchers will make a variety of strategic choices to maximize the impact of their research. In this context, “reinventing the Internet” should be interpreted very broadly. It is not restricted to innovation at any particular protocol layer, resulting for example, in a new version of IP. Instead, different researchers will answer the questions raised in Section 2 by developing new low-level protocols, defining different semantics for existing protocols, and creating new high-level services that run on top of today’s protocols. Similarly, “clean slate design” should be interpreted as a process, not an outcome. Researchers should not feel constrained by any of the assumptions of today’s Internet, but at the same time, we expect each research group to leverage one or more aspects of the existing Internet; not all will start from scratch. For example, a research group interested in network management might design new control protocols that work with today’s data plane, while a group interested in security might decide the only viable solution is to replace IP. Still another group might address security at the overlay level, leveraging most of the existing Internet.

While there are probably as many strategic approaches as there are research groups, discussions within the research community point to two broad perspectives, which roughly correspond to the questions raised in Section 2.2.2: Is the Future Internet best realized by changing the core architecture within the network, thereby solving existing problems and enabling new applications, or is change best achieved by creating new applications and services on top of the existing Internet, allowing these overlays to have a transformative affect on the Internet's core over time. We cannot answer this question today, but expect GENI to enable both perspectives.

4 Need for an Experimental Facility

A key element of any effort to redesign the Internet is a strategy for fostering the research *cycle*—drastically lowering the barriers that promising new directions developed by the research community face before transition to industrial development and deployment within the commercial Internet. This requires that we move well beyond the methodologies and facilities used today. An experimental facility that enables the research community to address the questions outlined in earlier sections must provide a seamless, end-to-end research process for taking ideas from conception, through validation, to deployment, similar to the idealized process shown in Figure 4.1.

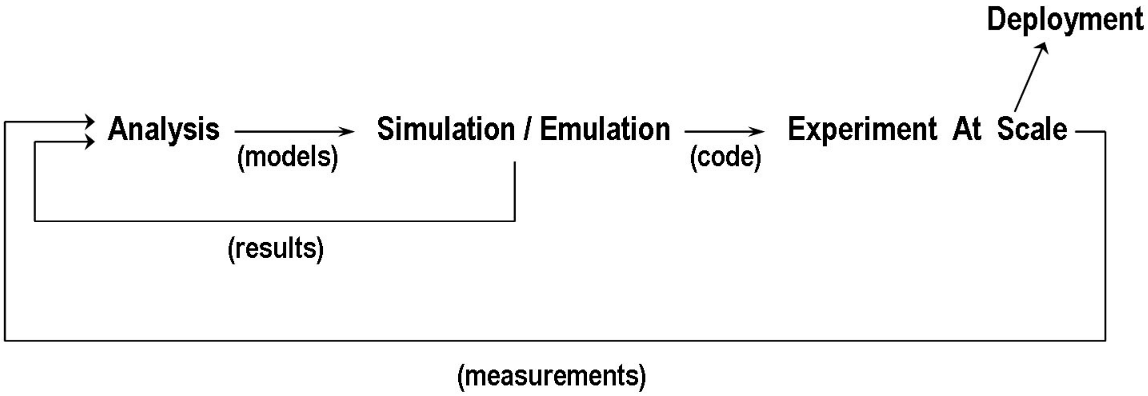


Figure 4.1: Seamless end-to-end research process, from early experimentation, through real-world validation at scale, to deployment.

Unfortunately, it is well known within the networking research community that we lack effective methodologies and tools for rigorously evaluating, testing, and deploying new ideas. As depicted in Figure 4.2, today researchers are able to simulate and experiment with small-scale prototypes, but are unable to perform experiments at a meaningful scale. This is not surprising considering the barrier-to-entry for experimenting with a new network architecture: a research group would have to arrange for dedicated hardware spread over a wide geographic area, acquire its own networking bandwidth, provide its own management and operational support, be responsible for its own security and fault isolation, implement its own measurement instrumentation, and so on. The consequence of this chasm is that standards bodies and early commercial adopters look with a skeptical eye towards any new networking idea backed solely by simulation results or small-scale experimentation. Therefore, we propose a new facility, called GENI (Global Environment for Network Innovations), which will radically improve the process by which research goes from the idea stage through validation to deployment. Building GENI is essential to the process of discovering the Future Internet.

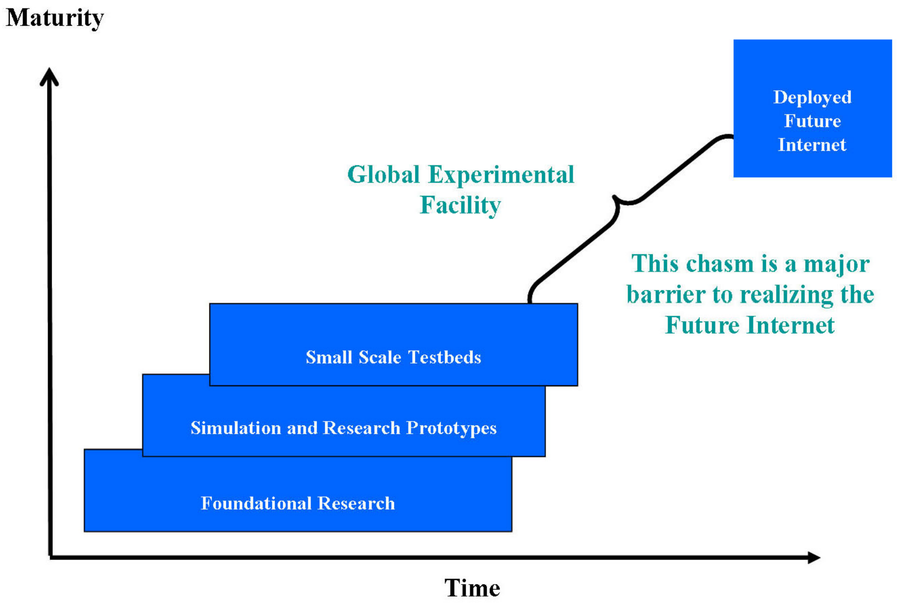


Figure 4.2: Chasm between small scale testbeds available today and the ability to adequately validate and deploy architectural innovations is a barrier to realizing the Future Internet.

4.1 Existing Methodologies

Today, most new ideas for how the Internet might be improved are initially evaluated using simulation and emulation. These techniques are invaluable in helping researchers understand how an algorithm or protocol operates in a controlled environment, but they provide only a first step. The problem is that today's simulations tend to be based on models that are backed by conjecture rather than empirical data; models that are overly simple in virtually every attribute of practical significance—topologies, administrative policies, workloads, device failures, and so on [FLO02, FLO01].

To truly understand complex protocols and comprehensive network architectures demands extensive live experimentation. Toward this end, testbeds are the conventional second step in the research process. However, current testbed paradigms are inadequate to this task, largely due to what the community refers to as the *testbed dilemma*. Traditional testbeds can be roughly categorized as production-oriented or research-oriented [AND05]. Production testbeds, such as Internet2 [I2], support real traffic from real users, often in large volume and across many sites. As such, they provide valuable information about the operational behavior of an architecture. However, the users of such a production testbed have no choice about whether or not to participate in the testbed and usually do not even realize that their traffic is part of an experiment. They thus expect the performance and reliability to be no worse than the standard Internet. Production testbeds must therefore be extremely conservative in their experimentation, using well-honed implementations of incremental changes.

Research testbeds (such as DETER [DET]) do not carry traffic from a wide variety of real users but instead are typically driven by synthetically generated traffic and / or a small collection of intrepid users. This allows them to be much more adventurous, capable of running first-cut implementations of radically new designs. Unfortunately, this lack of real traffic also renders the results much less indicative of real operational viability. As a result, neither kind of testbed—production or research—produces the data needed to adequately evaluate new architectures. It is therefore difficult to make a compelling case for new architectural designs based on a testbed evaluation.

A second limitation to traditional testbeds is that the community must know what testbed it wants to build. Consider that if the goal of this research were narrowly conceived as the development of exactly one proposal for a new architecture, and the subsequent demonstration of that proposal, then one approach to experimental infrastructure would be a purpose-built testbed targeted to that single proposal. However, this view does not reflect the reality of how this work will proceed. There is no reason to believe that there will be only one proposal for a new design. The best ideas will emerge from a competition among different proposals. Ideas may merge and diverge, and different options will require testing and validation. At any moment, there will be different ideas with different maturity and reflecting different approaches. This means that an experimental infrastructure, to be useful, must support multiple simultaneous trials and demonstrations. A primary feature of such a facility is that it can support multiple experiments at the same time.

Over the last several years the network research community has addressed this second limitation by building and using sharable, general-purpose emulation facilities. The most notable examples are Emulab [EMU, WHI02], Orbit [RAY05a], and WAIL [WAIL], which provide a room full of network equipment that can be remotely configured for different experiments. Emulab and Orbit support full programmability (i.e., run researcher-provided code), plus the ability to construct different topologies and different interference models, while WAIL allows commercial network gear to be parameterized and configured for different experiments. While these emulation environments allow experimentation with actual protocol implementations, they suffer from the same limitation as simulation in that they run only synthesized workloads and they are not available to end users.

However, the community is optimistic that a general-purpose facility that supports real-world experimentation at scale, yet resolves the testbed dilemma, is feasible. This is based on recent experience with a wide-area platform for evaluating and deploying overlay networks. The platform, called PlanetLab [PET02, BAV04], began as a grass-roots effort in mid-2002. It currently consists of over 635 nodes distributed across 300 Internet sites, and supports 425 network and distributed systems research projects. Each project acquires a “slice” of PlanetLab’s global resources, in which it runs an experimental overlay network. Some experiments run for a limited length of time, but many run continuously, with users “opt’ing in” to overlays that offer valuable services, thereby stressing these experimental services with real traffic and workloads. Users opt-in because network services running on PlanetLab provide value above and beyond the current Internet. Today, over one million unique IP addresses (client and server machines) send traffic through PlanetLab-provided services.

While PlanetLab serves as a prototype of the sort of facility we imagine, it does not adequately meet the needs of the research community. First, PlanetLab consists of a set of commodity PCs connected by today’s Internet. To fully explore the agenda outlined in Section 2 requires a much richer set of node and link technologies, especially with respect to wireless networks. Second, PlanetLab is grossly under-provisioned, both in terms of the capacity of the cycles, memory, and storage available to each research group, and also in the excessive bandwidth load it places on hosting institutions. Third, while PlanetLab is designed to support overlay research “on top of” today’s Internet, it does not adequately support experiments “inside the network”. Finally, PlanetLab provides a minimal set of support services, mostly designed and supported by graduate students, making the learning curve for research groups much higher than it should be.

4.2 Goals and Scope

The end goal is clear: we want to support a research process that provides a smooth path from concept to practice. Simulation and emulation must be augmented with live systems that can be deployed on a wide-area facility, allowing for real users and real-world experiments at scale. This facility must also be heavily instrumented to produce the measurements needed as input to the next round of simulations. Because applications, services, and architectures running on the facility are operating at large scale and in the real world, there is an increased likelihood that successful ones will make the transition to wider deployment, possibly even using the same infrastructure technology as the facility itself. Without such a facility, research ideas will stay in the ephemeral state, never validated to the degree of realism or the scale needed to convince industry or standards bodies.

Toward this end, GENI must provide an environment in which multiple new network architectures and services can be deployed, placing as few restrictions as possible on the experimental architectures and services that operate on the facility and on the capabilities provided by the facility. GENI must include a diversity of link and node technologies, and permit connection of arbitrary edge devices and networks. GENI must be designed to bridge the gap between production testbeds (which constrain research), and research testbeds (which constrain users). It must be capable of attracting and supporting users of its services beyond the research community. This is essential for allowing new innovations to be evaluated at scale, and for creating a population of users whose demonstrated interest in a new capability can stimulate technology transfer to the commercial sector.

To meet these goals, GENI will provide an environment in which a diverse set of experimental networks—each with its own distinct architecture—can operate. GENI constrains the hosted activities to the minimum extent possible, and provides for varying degrees of isolation and interconnection among these activities. The common part of GENI, which we refer to as the *GENI substrate*, provides the mechanisms for allocating and configuring resources and ensuring the necessary isolation.

GENI should be viewed as a dynamic artifact: the physical resources, management capabilities, implementation, and even the substrate design will evolve over time. The physical resources in GENI will include a mix of dedicated physical links and nodes, virtual components contributed on a permanent or temporary, and diverse edge systems such as wireless and sensor networks. The substrate will incorporate standard service policies and interfaces to enable organic growth, provide incentives to new communities to contribute, and manage dynamic resources available to the substrate on a temporary basis under various terms.

4.3 Key Concepts and Requirements

Several key concepts will play a central role in the design of the GENI, which consists of a collection of substrate resources, including nodes, links and edge subnets. Each experiment will run on some subset of the GENI resources. We call the substrate resources bound to a particular experiment a *slice*, borrowing the term from PlanetLab. 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 substrate will include management software that is used to allocate resources to slices, and ensure that slices do not interfere with each other.

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

We also note that there will be two different usage models for GENI slices. In the first, 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, 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.

In summary, GENI has the following requirements:

- **Service/architecture neutrality:** What is most important for research in network architecture and services is that the level of abstraction be low enough to permit full experimentation. Different slices of the GENI may support different experiments at the same time

- **Edge diversity:** GENI should enable heterogeneity in the end systems that connect to it and participate in the experiments running within it. In particular, it should enable the connection of limited functionality end-systems (such as wireless PDAs and sensors) connected by a variety of technologies (such as wireless and sensor networks).

- **Ease of user access:** Mechanisms are needed to make it easy for users to join one or more experimental services running in GENI, and to transparently fall back to the legacy Internet whenever the experimental network cannot provide the requested service.

- **Global reach:** To support experimentation at scale, and to maximize the opportunity to attract real users, GENI must have as wide of reach as possible. Access cannot be limited to only those few sites that host backbone nodes.

- **Instrumentation and data analysis:** The GENI substrate, along with all the architectures and services deployed on it, must be heavily instrumented. The generated data must be collected and archived, and analysis tools developed.

- **Federation and sustainability:** To ensure the sustainability of GENI, it should be possible for participating institutions to contribute resources in return for access to the resources of the GENI as a whole. In general, it should be possible for other research communities to ``opt-in'' by connecting their purpose-built networks (including dedicated transmission pipes and sensor networks) into the GENI substrate and running their applications and services in a slice of GENI.

- **Inter-slice composition:** GENI must enable interconnection among slices by mutual consent, and between slices and the legacy Internet. This permits slices to host network services with external users, and/or to act as transit networks. Nothing should prevent a researcher from inter-connecting a virtual network running within a slice with another network. This other network could be running within another slice of GENI, or it could be the legacy Internet or another custom network (or testbed) that runs over standard IP protocols.

- **Policy and governance:** Since GENI will comprise shared infrastructure, there must be a governance process to guide allocation of resources to slices, and a software architecture that implements and enforces the policies. Some slices will likely require strong performance isolation, which will make some form of admission control necessary.

4.4 Success Scenarios

By the time the construction and operational life of GENI ends in roughly 20 years, how will we know if we have been successful? It would be easy for us to say that by that time we will have developed and deployed a Future Internet that will have completely replaced today's Internet, fixing all of its problems and providing society a firm foundation for the distributed applications that will by then be in widespread use. However, as scientists, we must admit that the path to the future is not likely to be so direct. A rule of thumb of technology innovation is that (1) it usually takes much longer than one might predict at first for any fundamental innovation to reach widespread use, and (2) once in widespread use, a fundamental innovation will eventually have much more profound benefits than even the most optimistic might predict at the start of the process. This pattern has held for many of the innovations we have come to rely on today. For example, who could have imagined when Marconi invented the first practical long distance radio that eventually a third of the people on the planet would have personal cell phones providing them the ability to communicate instantaneously with nearly anyone anywhere on the planet, all based on the same principle? Or if they had imagined such a future, that it wouldn't happen immediately, but rather would take over a hundred years to come to fruition? The Internet itself is over thirty years old, and for most of that time, it appeared to be of mostly academic interest and use; the potential benefits of the Internet for society are only just beginning to be seen. It is precisely to put the Internet in a position to fully reach those benefits that we have proposed GENI.

Thus, we believe a better test than widespread use of a Future Internet will be that we will have created a path to widespread use. One of our goals is to create a robust and trustworthy foundation for future applications, so that society can start to rely on the Future Internet. It is hard to imagine the Internet reaching its potential without this capability. An indication that we are on that path will be the development of new applications on a Future Internet that will only be possible on a robust and secure framework. For example, a significant barrier to fully leveraging the Internet for reducing health care administrative costs is the challenge, in today's environment, of keeping digital medical records both private and free from malicious harm. This is meant as just an example; many, many other current and potential uses of the Internet are sustainable only when placed on a reliable and secure foundation.

Given that GENI is designed to support as wide of set of proposed network architectures as possible, there may not be a single architecture as there is in the current Internet. There are many ways to deliver the service users want, and the key to progress is likely to be to enable continuous competition among these different approaches. A test for the success of the Future Internet is whether it is dynamic and evolving in its protocols and services. The ability to support these multiple coexisting systems then becomes the crucial universal piece of the architecture.

Further, success may come in multiple flavors. By demonstrating the value of a new architecture, particularly for demanding applications, we may facilitate the modification of the existing Internet to incorporate a set of new ideas and functionality. Or, it may be that by developing a set of advanced services well beyond what the current Internet can easily provide, users come to rely on these services, rendering the underlying Internet less and less important or visible to end users. Or it may be that a ``next generation'' architecture, after having been validated on GENI, would, through some magical process of consensus and daring, be adopted by ISPs and router vendors alike.

To us, an *organic* deployment story seems most likely. In this organic story, there is no discrete or global decision point at which the old world accepts and incorporates the new technology; the process is continuous and incremental by definition. The players that represent the old order may respond to market opportunities, for example, by providing high-performance or more cost-effective implementations of the new technology demonstrated on GENI. Simultaneously, the uses that rely on the unique characteristics of GENI—its security, reliability, and flexibility with respect to new application domains—could over time become more and more prevalent, so that increasing numbers of users have their Internet use mediated by GENI itself, or by services originally launched on GENI.

As improbable as this organic story may sound, there is at least one existence proof that it works: the Internet itself. Both the original ARPANET and the Internet that followed began as overlays running on top of the entrenched telephony system. The disruptive Internet technology eventually transformed the underlying telephony system from being circuit-based to being packet-based. Today, it is difficult to say where the old technology ends and the new technology begins.

5 GENI Design

The sections up to this point provide the rationale for GENI: they outline the research agenda and argue that a new facility is needed to undertake this agenda. This section describes the

GENI facility itself, with the goal of introducing the key aspects of its design. Section 5.1 gives a high-level overview of the design, with Sections 5.2 and 5.3 going into detail on the various hardware and software components, respectively. Section 5.4 then discusses other design considerations and 5.5 summarizes the unique capabilities that GENI provides to address the needs of the research community.

5.1 System Overview

To realize the vision of GENI serving as a catalyst for networking research and early-stage deployment, it must be possible for researchers to assemble a widely distributed set of resources into an experimental network. Researchers do this by acquiring a slice of the GENI substrate, and running the experimental network application, service, or architecture in that slice. The resources that make up the GENI substrate must span the full spectrum of existing and imagined network technologies, so as to not overly constrain the virtual networks that researchers can build. This requirement introduces a significant complication into the design of GENI, because no single category or class of existing hardware can meet it. Further, as new technologies are developed and mature, the bounds of what constitutes a network are stretched, and any static, rigid deployment of infrastructure for GENI risks becoming obsolete.

To meet these requirements, the basic design of GENI is divided into two parts: (1) a *physical network substrate*, and (2) a *global management framework*. This section briefly introduces both parts; they are described in more detail in Sections 5.2 and 5.3, respectively. The central concept that ties these two halves together is the idea of slice*: The management software virtualizes the physical substrate so that it can be shared among multiple network experiments, with each experiment embedded in a slice of the physical substrate.*

5.1.1 Physical Network Substrate

The physical substrate consists of an expandable collection of *building block* components. While GENI is designed to allow new building blocks to be added over time to reflect changing needs and new technology, we propose an initial set of building blocks; they should be interpreted as a snapshot of GENI's physical substrate as of today. This initial set can be divided into three broad categories: one or more *node* technologies, a variety of *wireless subnet* technologies, and a mix of *link* technologies.

The GENI substrate will be configured to include a nation-wide backbone network with at least one lambda (10Gbps) of capacity available between one to two dozen Points-of-Presence (PoP). Each PoP, in turn, will host a high-speed forwarding node, where we envision two candidate node technologies: one based on customizable high-speed hardware, and a second based on emerging optical switches. Edge sites (e.g., university campuses) will host GENI nodes with significant compute and storage resources (i.e., clusters of commodity PCs). These edge sites will connect to the nearest backbone PoP using the most appropriate tail circuit technology—e.g., MPLS or FrameRelay circuits with 45-155Mbps of capacity—as well as by tunneling through today's Internet. These edge sites will not be limited to the U.S., but will also include international sites, giving GENI global reach, and thus supporting experiments running at scale. Additional edge sites will host wireless subnets of different types, including urban 802.11-based ad hoc meshes, suburban subnets based on 3G and WiMax, sensor networks, and subnets built around cognitive radios. Finally, the national backbone will be connected to the legacy Internet by connecting the nodes at one or more PoPs to an Internet Exchange (IX), thereby providing connectivity to multiple commercial ISPs.

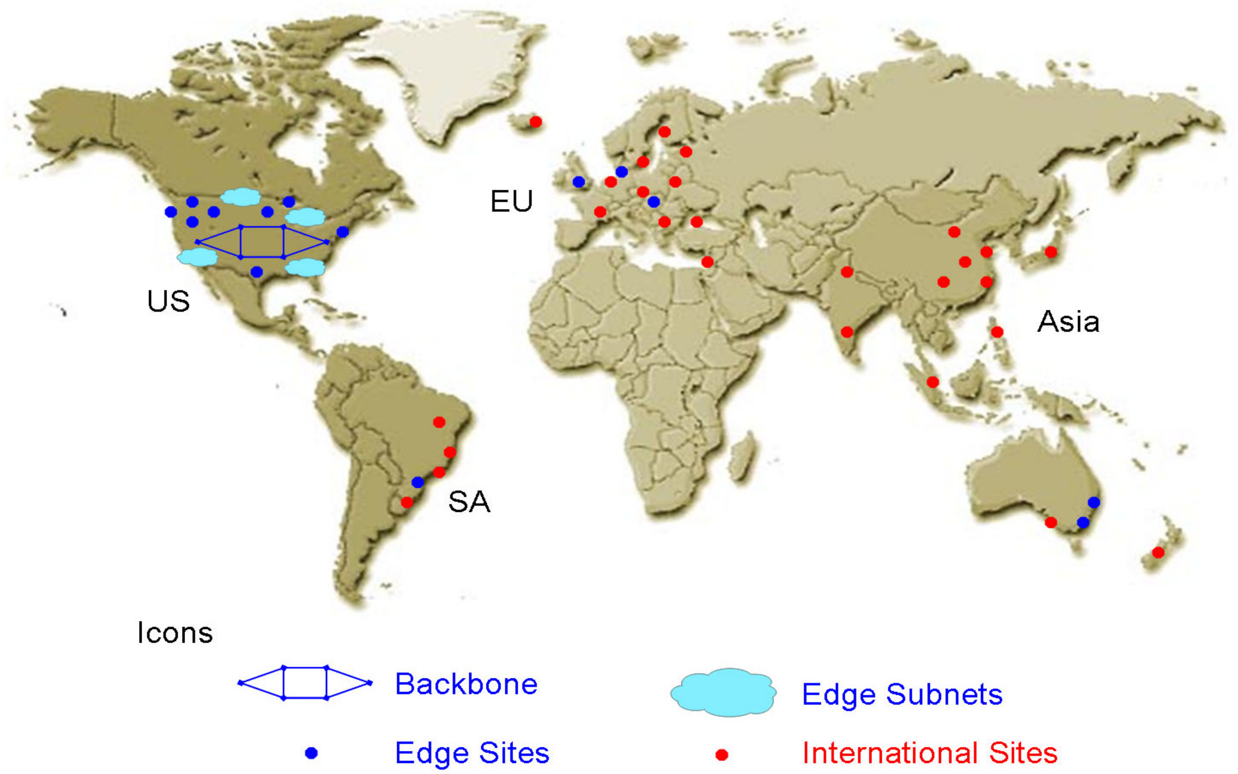


Figure 5.1: GENI includes both wired and wireless edge sites around the world connect to a high-capacity U.S. backbone. We expect many international sites (shown in red) to be contributed by other participants that are federating with GENI.

Figures 5.1 and 5.2 illustrate the design of GENI. Figure 5.1 shows GENI's global reach, with edge sites around the world connecting to a high-capacity backbone via PoPs in the United States. As discussed in Section 6.8, we anticipate that outside organizations and governments will participate in GENI as well; the red dots in Figure 5.1 represent the nodes they contribute. Figure 5.2 focuses on a single PoP, where (clockwise, from top right) a sensor network, a traditional wired network of workstations, and a wireless network connect to the GENI backbone. Packets flow between GENI nodes located on each network and the high-speed forwarder at the PoP over circuits (fat blue lines) or tunnels (thin black line). The forwarders at various PoPs communicate via the national backbone.

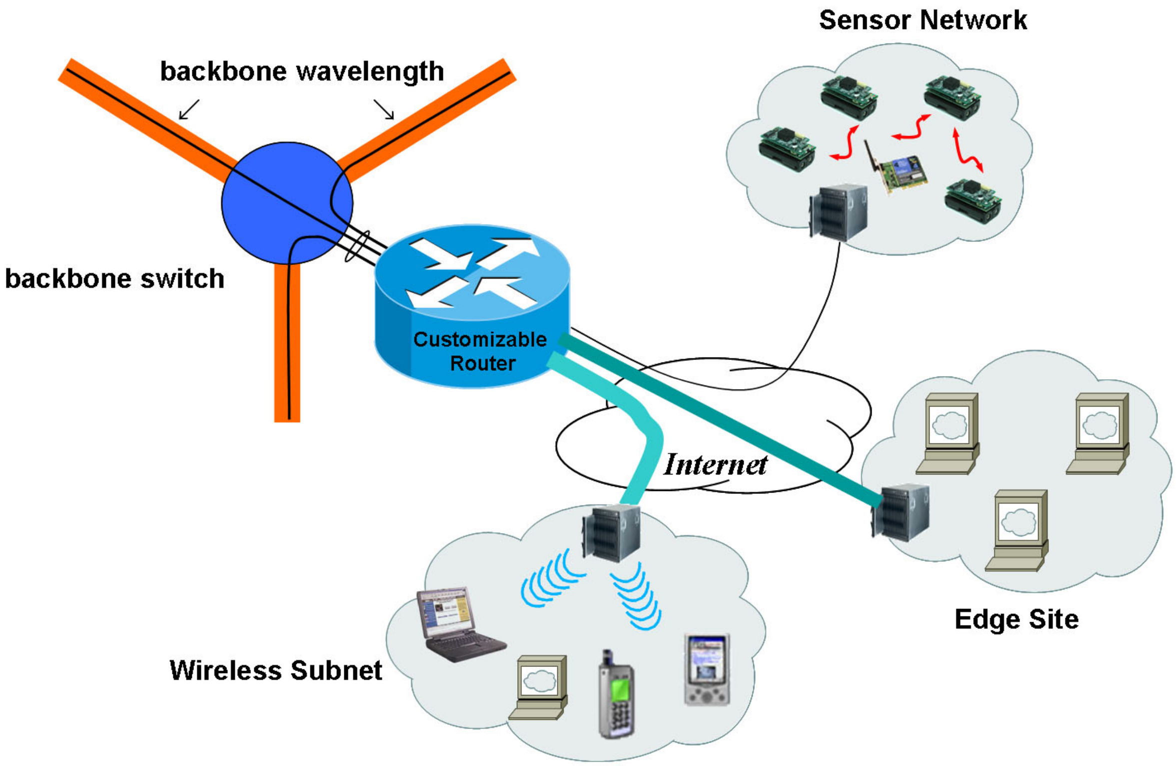


Figure 5.2: Detailed view of a single backbone PoP; a GENI high-speed customizable router connects edge sites to the GENI backbone. Edge sites are connected by a variety of tail circuit technologies, including MPLS circuits and tunnels through today's Internet.

The justification for this global configuration of building blocks is straightforward. A national backbone of at least 10Gbps capacity is required to support research in network design (e.g., routing, failure recovery, network management, and so on), to validate solutions at realistic forwarding speeds, and to carry traffic between edge systems. The clusters running at edge sites serve two purposes. First, they act as "ingress routers" for local users wishing to take advantage of architectures and services running in the backbone. Second, they serve as overlay nodes that host broad-coverage network services and applications that require many points-of-presence through the network. Having edge nodes at hundreds of sites allows these experiments to run at scale. The set of wireless subnets span the spectrum of available and emerging technologies, and connecting these subnets to the backbone permit research on end-to-end connectivity. Finally, we connect the backbone to the commodity Internet to allow users to access legacy content and services via GENI. Without this capability, no users will use the experimental services and architectures deployed on GENI, dramatically limiting its research value. Section 5.3 describes and justifies the specific building blocks we elect to include in GENI.

5.1.2 Management Framework

The second major part of GENI, the management framework, knits the building blocks together into a coherent scientific instrument—a single global-scale facility that is capable of supporting the research cycle outlined in this document. The essence of the management framework is its support for the slice abstraction. It is primarily implemented in software, and it 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.

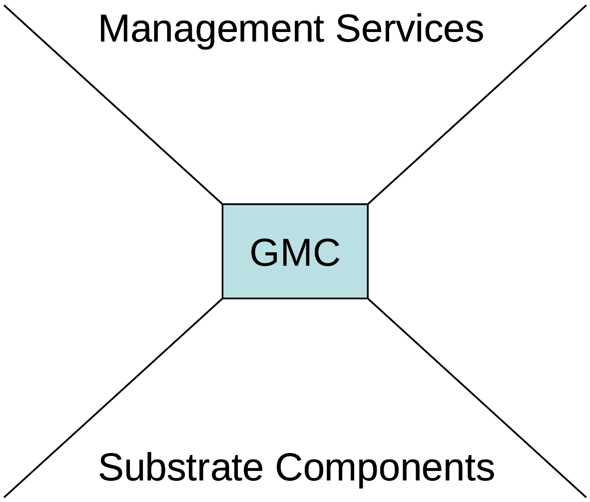


Figure 5.3: Overview of the GENI architecture, where a minimal core, the GENI Management Core (GMC), logically connects a wide-range of management services with a diverse collection of building block components.

The key to providing such a 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. The resulting architecture is depicted in Figure 5.3, where the core, its support for a set of high-level services, and its support for a set building block components are all described more fully in Section 5.3.[1]

5.2 Building Block Components

GENI's physical substrate consists of nodes that run experimental applications, services and network architectures, along with the underlying physical layer communication media that connect them. The primary goal of the substrate is to support the multiple simultaneous experiments, each running in their own slice of GENI resources, allowing each slice to support its own network architecture with different ways of providing naming, addressing, forwarding, routing, security, management, and so on. The power of the GENI substrate lies in the diversity of technologies for the nodes and communication media, and the ability to virtualize these resources so they can be shared across different experiments.

This section identifies the building block components that we will use to construct GENI, adding further definition to the overview presented in the previous section. The description both identifies candidate technologies for each component and discusses the component's ability to support virtualization and programmability. The building blocks are of three broad classes: (1) three node technologies, (2) two categories of link capacity, and (3) five wireless subnets.

5.2.1 Flexible Edge Device

Clusters of commodity PCs will be the workhorse nodes in GENI. They provide the computational resources needed to build wide-area services and applications, and even in situations where special-purpose hardware is more appropriate, general-purpose processors will allow researchers to work on the functionality of new architectures while these new technologies are developed and hardened. In particular, PC clusters will be distributed to 200 edge sites, where they will host overlay services and serve as "ingress routers" for new network architectures. We will vary the size and unique capabilities of these clusters as user demand dictates, for example, by adding large storage capacity to a subset of the edge sites.

The PC in each cluster will run an operating system that supports one or more virtual machines, each of which is bound to some subset of the PC's memory, disk, CPU, and network capacity. Each slice that wishes to run on such a node is allocated a virtual machine on one of the PCs. A virtual machine is programmable in the most elemental sense, in that a researcher could create and run one or more computer programs in each virtual machine in its slice. For maximum flexibility, researchers can write their own software from scratch, using conventional programming languages; over time, we envision that researchers would create software modules that provide key services that others can use to build their prototypes.

---

[1] Although GENI is designed to support research with new network architectures, GENI itself has an architecture, which is to say, it can be factored into a set of interacting components, each of which supports a well-defined interface. Whether we are referring to GENI's architecture or an experimental network architecture that runs on top of GENI should be clear from the context.

To support the abstraction of a virtual machine, the operating system running on the underlying computer must (1) allocate and schedule resources (e.g., CPU, bandwidth, memory, and storage) so that the runtime behavior of one slice does not adversely affect the performance of another running on the same node, (2) partition or contextualize the available name spaces (e.g., network addresses and file names) to prevent a slice from interfering with another, or gaining access to information in another slice, and (3) provide a stable programming base that cannot be manipulated by code running in one slice in a way that adversely affects another slice (e.g., slices would not be given root or system privilege).

Virtual machines are now a mature technology and they have proven effective in supporting experimentation with distributed services in PlanetLab [BAV04, BAR03], making them a natural, low-risk starting point for building nodes for GENI. Today, user-level packet forwarding running in a slice can forward data packets at nearly 1 gigabit/second. As GENI evolves, we can move key functionality into the operating system for faster throughput and provide an increasingly fine granularity of sharing of the system resources by exploiting hardware support for virtualization [IVT].

5.2.2 Customizable High-Speed Router

Although PCs supporting virtual machines are a natural starting point for a node technology, speed and the ability to interface to diverse communication media rapidly become limiting factors. Moving beyond commodity processors, GENI will also include customizable high-speed routers consisting of multiple programmable processing elements, as well as line cards for terminating the physical links and directing traffic to/from the appropriate processing elements via the switching fabric, as shown in Figure 5.4. Depending on the capabilities of the processing elements, a slice might have dedicated access to an integral number of elements or run on a virtual machine allocated part of the resources of a single processing element. As with the flexible edge devices, the customizable router is fully programmable, and each slice could run separate customized software on its dedicated computing resources.

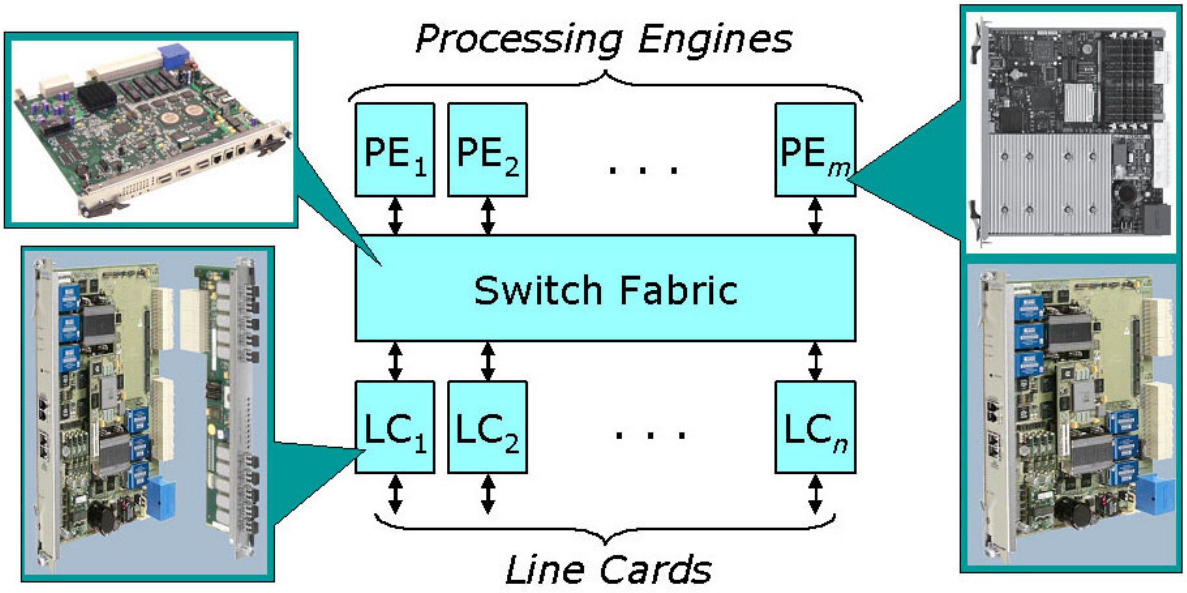


Figure 5.4: GENI customizable high-speed router consisting of an internal switching fabric that connects a heterogeneous collection of processing engines (e.g., general-purpose processors and network processors) and multiple with network line cards.

Building a customizable high-speed router is made possible by two important innovations in recent years. First, the emergence of standards for the interfaces between components in a router subsystem (e.g., the Advanced Telecommunications Computing Architecture) makes it possible to connect the processing elements into a standard chassis with standard backplane, power distribution system, and cooling fans, obviating the need for network architects to handle low-level physical and electrical issues (e.g., cooling and clock distribution). Second, although the processing elements could be conventional microprocessors, advances in network-processor technology allow use of specialized programmable devices that include high-performance I/O and multiple processor cores. This gives us two options with different trade-offs between network performance (faster with network processors) and ease of software development (simpler with conventional microprocessors). Over time, we also envision that router vendors would offer routers with native support for virtualization. These will not be fully open in the sense that researchers can change the underlying software, but they are expected to expose low-level interfaces to support new services. Our plan is to investigate the use of both styles of customizable routers in GENI.

Customizable high-speed routers will also need to run an operating system that virtulizes the raw hardware. We expect to leverage the same OS support as runs on edge devices, modified to provide access to network devices. That is, the OS will “lower” the level of virtualization from the socket layer to the device layer. The OS will likely also be extended to allocate complete processing elements rather than just virtual machines.

5.2.3 Dynamic Optical Switch and Network Layer

Technology is moving beyond the traditional notion of connecting a sequence of routers by a wavelength at a fixed bit rate, with multiple of these disjoint wavelengths bundled onto a fiber. Today's technology allows us to dynamically utilize the optical bandwidth in conjunction with switching at the electronic router level. Using new control plane technology (e.g. GMPLS, BGB), and standardized labeling schemes (MPLS), routers can directly interact with, and control, provision-able optical bandwidth via optical switches. We envision both a short-term and a long-term building block that exploits this capability.

The simplest and nearest term optical switch technology utilizes circuit-based optical add / drop multiplexers and optical cross-connects to configure the wavelengths as circuits between relatively arbitrary router interfaces across a network. No longer does traffic need to pass through every router, but instead, the interface is connected to a traffic-engineered wavelength (circuit), and that wavelength and its traffic can optically bypass intermediate routers until terminated on the destination router. This approach significantly saves on the amount of router hardware (power and space) needed to support a scalable network. By making these optical add / drop multiplexers and switches reconfigurable—hence the terms *reconfigurable optical add/drop multiplexer* (ROADM) and *photonic cross-connect* (PXC)—the control plane can be used to set-up and re-route optical circuits on demand. To achieve this end goal, the management system becomes a complex entity that must keep track of all physical resources, connection patterns, route and state (node and line) tables, optical regeneration points as well as physical layer transport rules. This is something that has to date not been fully realized in any network, but is being heavily pursued by industry and other optical network initiatives in Japan, Europe and Asia.

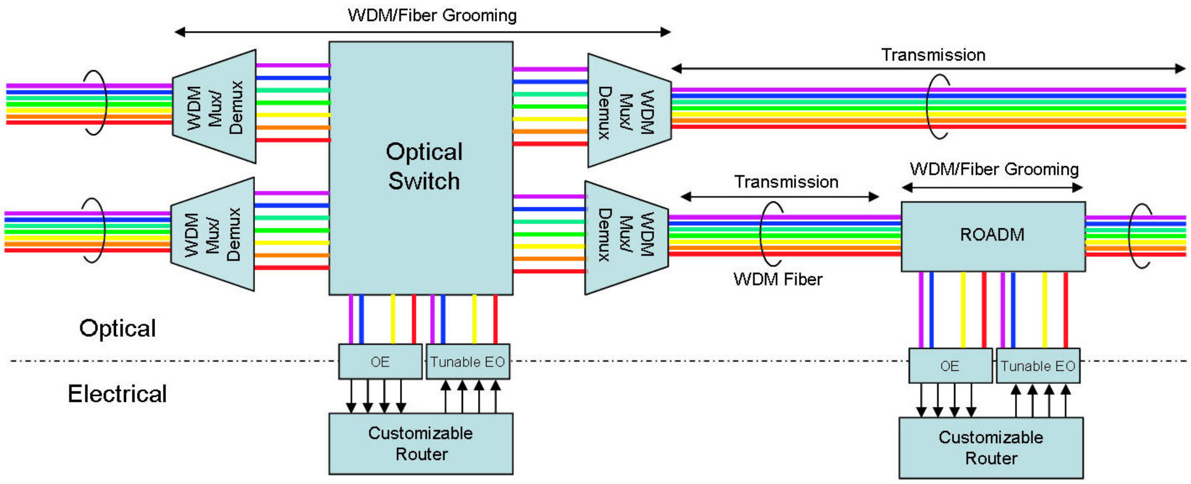


Figure 5.5: Optical fiber is configured with a Reconfigurable Optical Add/Drop Multiplexer (ROADM) that provides the attached with router fine-grain control over light paths across the optical fabric.

The type of optical device described above operates at the reconfigurable circuit level. In the first instantiation of GENI, this is the fist type of optical switching that will most likely be used, as shown in Figure 5.5. Bringing online the optical line cards, ROADMs and cross-connects, interfacing with programmable routers to a common control plane, all under the GENI management system will be a critical task. The programmable GENI routers will need to have the power and flexibility to reserve and access any available wavelengths as circuits, so the setup and tear down time under this model will be on the order of the circuit round trip propagation delay. Optical bandwidth can be used to multicast, send parallel instructions to logically adjacent nodes, perform simultaneous CP instruction cycle transfers and memory caching, and so on.

In the longer term, well beyond the reconfigurable optical circuit model, is the dynamically switched optical bandwidth model enabled by a new generation of rapidly configurable switching technologies (e.g. rapid tunable laser and wavelength converter based switches). These new switch technologies can be used to realize *packe*t add / drop multiplexers (PADMS) and fast packet switches. This type of node will allow the GENI processing and memory cycle transfers and requests to directly access the optical bandwidth on any available wavelength during a free time period. Networks of this type have been demonstrated at the proof of concept level by groups in the US and Japan. In such a network, circuits are not established ahead of time so fast media access and control techniques must be employed. In the foreseeable future, optical buffers will not be available to GENI, so a variety of methods will be employed for buffer-less optical packet transmission, where scheduling occurs at the edge and buffering occurs in electronics at the edge.

The optical switching technology supports virtualization by dividing capacity by wavelength or time slot, or both. As the network moves from dynamic circuit wavelength based to wavelength / time-dynamic, the degree of sliceability of optical network bandwidth will move from wavelengths on fibers to slicing time within wavelengths on fibers. Access to the optical network bandwidth will be a programmable construct like any other parameter in the network. There will be certain levels of programmability visible to the user, while other levels will be visible to the management software. The user will have access to which wavelength(s) and time slots at an ingress point to access and where the wavelengths or time slots will terminate. Through programmability, the optical network will become an extension of the programmable router as an extended backplane.

5.2.4 National Fiber Facility

GENI includes a national backbone network that, in turn, will be built on top of a national optical fiber plant. Ideally, this facility will include one to two dozen points-of-presence (PoP) interconnected by optical fiber, with at least one lambda (10 Gbps) allocated to GENI. At each PoP, we expect to connect the various GENI node technologies into a network switch, also at 10 Gbps rates. It is by connecting one of the three node types just described at these PoPs that we plan to construct a nation-wide network backbone.

The GENI management software can allocate portions of the 10 Gbps bandwidth to slices running on nodes connected at the PoPs. It is also possible that a single slice might be allocated a backbone topology consisting of dedicated lambdas between particular pairs of nodes. The infrastructure can support nodes that operate at the optical level, facilitating experimentation with new architectures that blur the traditional boundaries between the network and physical layers. The optical fiber plan is inherently virtualizable, since each lambda (or sub-lambda) can carry traffic independently of the other lambdas on the same link, essentially providing a constant-bit-rate service. Using an existing fiber plant infrastructure allows GENI to allocate substantial bandwidth to a slice, for evaluating new node technologies and experimental architecture at large scale.

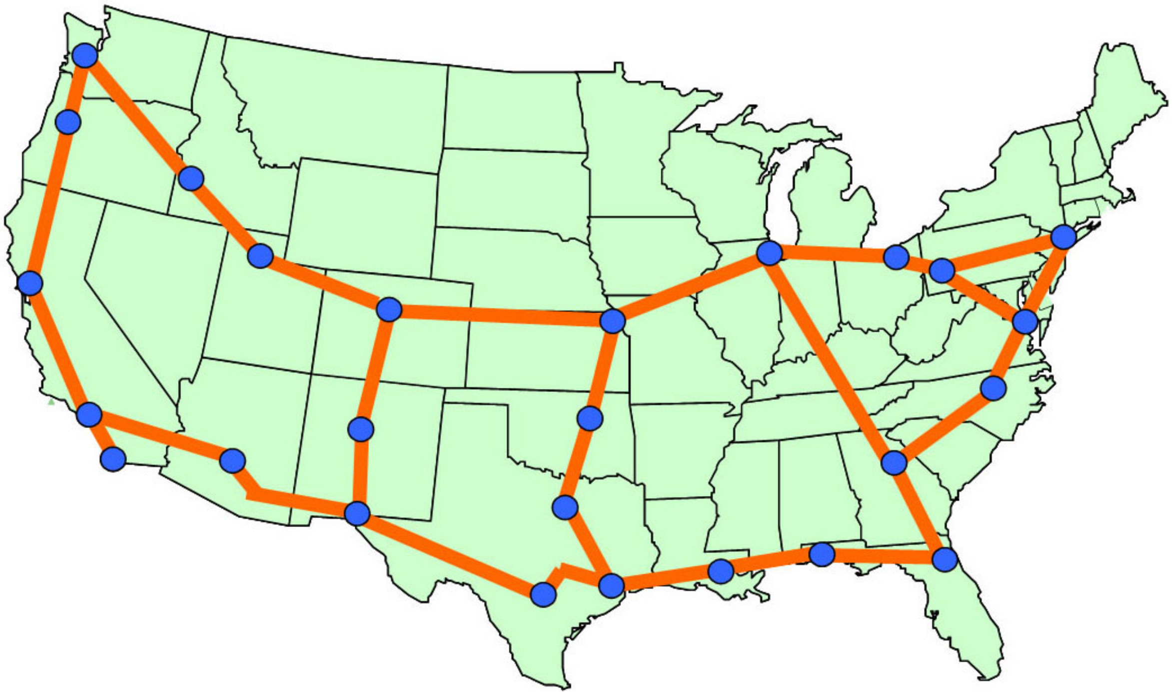


Figure 5.6: Current configuration of the National Lambda Rail (NLR), with includes 26 PoPs across the U.S.

Fortunately, multiple optical fiber facilities exist in the U.S. For example, the National Lambda Rail (NLR) is a nation-wide optical fiber plant that consists of 26 PoPs distributed across the United States connected by optical switches and fiber, as shown in Figure 5.6. Research efforts, such as GENI can request and be granted one or more lambdas (light paths) between the PoPs. The existence of the NLR infrastructure substantially lowers the risk for GENI to support experimentation with new ways for backbone and metro network architectures to relate to the optical level.

5.2.5 Tail Circuits

For connecting edge sites to GENI, we envision a variety of link technologies, including physical links (such as a wavelengths on a fiber-optic cable), MPLS circuits, Frame-Relay circuits, or IP tunnels. Many of these links will connect edge sites to the backbone, but some will provide connectivity into the commodity Internet, again through a variety of link technologies (e.g., DSL and cable modem). Because any single backbone has a limited reach, it is largely by connecting sites via tunnels through today's Internet that GENI achieves global scale.

Virtualizing a given link technology requires a way to allow different slices to share bandwidth without interfering with each other (if necessary), to ensure the integrity of the concurrent experiments with different network architectures. Time-division multiplexing allocates a certain fraction of the time slots to each slice; for example, if one slice is allocated 60% of the bandwidth and another is allocated 40%, the node would transmit three packets of the first slice for every two of the second. In frequency-division multiplexing, the underlying media may have multiple wavelengths (e.g., for optical links) or frequencies (e.g., for wireless lengths) that can be used to transmit data. Each slice could be allocated specific wavelengths or frequencies for transmitting data from one node to another, without interfering with other slices transmitting on different wavelengths or frequencies.

Both techniques for subdividing bandwidth resources are in common use in today's networks, making them appealing mechanisms to incorporate into GENI. The chosen technique for virtualizing the link depends on the underlying media and the sophistication of the equipment. In addition, some slices may need a hard guarantee on link bandwidth and delay, whereas best-effort service may suffice for others. Slices that require only best-effort service can share bandwidth more freely. For example, when one slice is idle another slice could send more data. However, slices that require hard guarantees for complete isolation from other experiments should receive their share of the bandwidth independent of the behavior of other slices. In some cases, the underlying link technology may support basic forms of programmability through configuration. For example, a frame-relay or ATM circuit might support different kinds of quality-of-service guarantees, whereas an IP tunnel might offer just best-effort service depending on the underlying path and the cross traffic.

5.2.6 Urban 802.11-based Mesh Subnet

The first wireless subnet**,** an urban 802.11-based mesh / ad-hoc network, is designed to support real-world protocol experience with emerging short-range radios (see Figure 5.7). Ad-hoc mesh networks represent an important area of current research and technology development activity, and have the promise of providing lower-cost solutions for broadband access particularly in medium- and high-density urban areas. While protocols for ad-hoc mesh networks have been maturing, the research community has limited field experience with large-scale systems and application development. Research topics to be addressed using the GENI system include ad-hoc network discovery and self-organization, integration of ad-hoc routing with core network routing, cross-layer protocol implementations, MAC layer enhancements for ad-hoc, supporting broadband media QoS, impact of mobility on ad-hoc network performance and real-world, location-aware application studies.

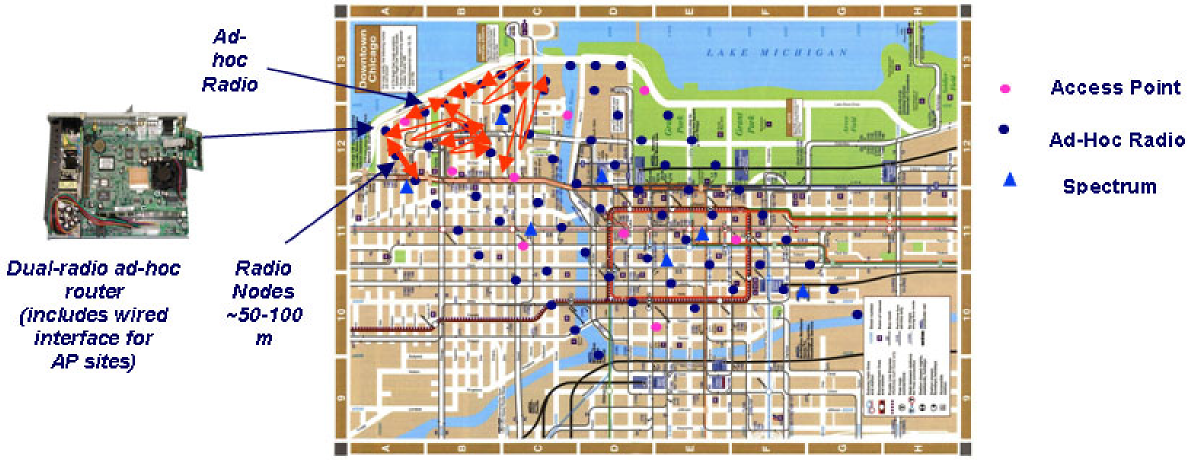


Figure 5.7: Example deployment of an ad-hoc 802.11-based mesh in an urban area.

The proposed ad-hoc mesh network in GENI will consist of ~1000 open API radio routers or forwarding nodes densely deployed in two or more urban areas or campus settings with coverage area ~10 Sq-Km. A typical node will be installed at ~8m height using available lighting poles or other utilities, and will have electrical power and a remote management interface. Approximately 25% of the nodes will be designated as access points with wired interfaces (typically VDSL or fiber) to GENI access routers. The deployed network will also support location determination via radio triangulation methods, and this information will be made available to experimenters as a service within the GENI software. End-users of the GENI system will be able to select subsets of wireless nodes for an experiment and deploy their own layer 2 (medium access and data link control) and higher layer (routing, multicast, overlays, etc.) protocols on these nodes using the software framework described in Sec. 5.3. The network will support experimentation with a variety of mobile computing devices including commercially available laptops, PDA’s and media devices with 802.11 interfaces.

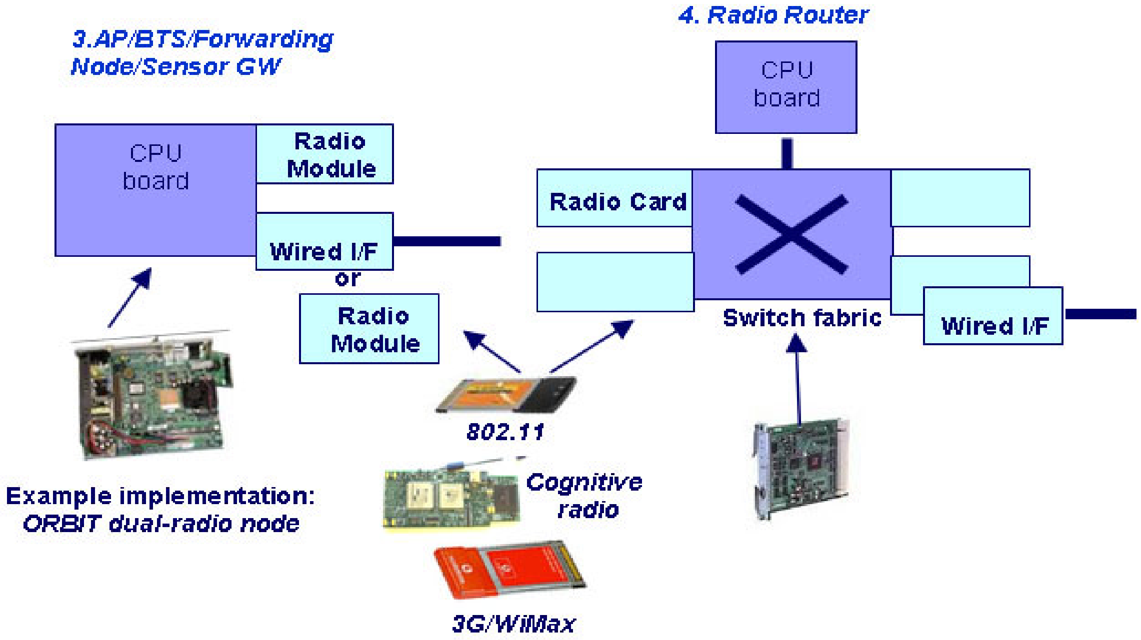


Figure 5.8: Schematic of programmable radio access points and radio router platforms.

Nodes in the network will be programmable, and will support applicable forms of slicing and virtualization corresponding to the capabilities of platforms used. A platform suitable for use in this experimental system is a dual radio processing node (see Figure 5.8) with the capability of bridging or routing between multiple radios (for example, IEEE802.11b and 802.11a) or between a radio link and wired network (for example, between 802.11a and Ethernet), and can thus be used as a general-purpose ad-hoc routing node, sensor gateway or wireless access point. The dual radio node used in the ORBIT testbed or the Intel Stargate board is an example of such platforms. Network virtualization across multiple radio technologies or a single radio technology and multiple frequencies is readily achieved with this platform. Processor and memory resources may be sliced using techniques discussed earlier for programmable routers. A second platform that may be considered for improved flexibility is a radio router with switching fabric and multiple radio and wired ports (see Figure 5.8). This platform can support a higher degree of slicing and virtualization using multiple radio technologies (such as 802.11a and 802.11b) or via non-interfering frequency assignments. The switching fabric also allows for attachment of multiple processing engines which can be used as dedicated protocol processors for virtual networks where desired.

5.2.7 Wide-Area Suburban 3G/WiMax-Based Subnet

The second wireless subnet is a wide-area suburban wireless network with open-access 3G / WiMax radios for wide-area coverage along with short-range 802.11 radios for hotspot and hybrid service models (see Figure 5.9). This wireless scenario is of particular importance for the Future Internet as cellular phone and data devices are expected to migrate from vertical protocol stacks such as GSM, CDMA and 3G towards an open Internet protocol model. Experimental research on future cellular networks and their integration into the Internet is currently restricted by the lack of open systems that can support new types of protocols and applications. Research topics to be studied using the proposed experimental network include transport-layer protocols for cellular, mobility support in the Future Internet, 3G / WLAN handover, multicasting and broadcasting, security in “4G” networks, information caching / media delivery, and location-aware services.

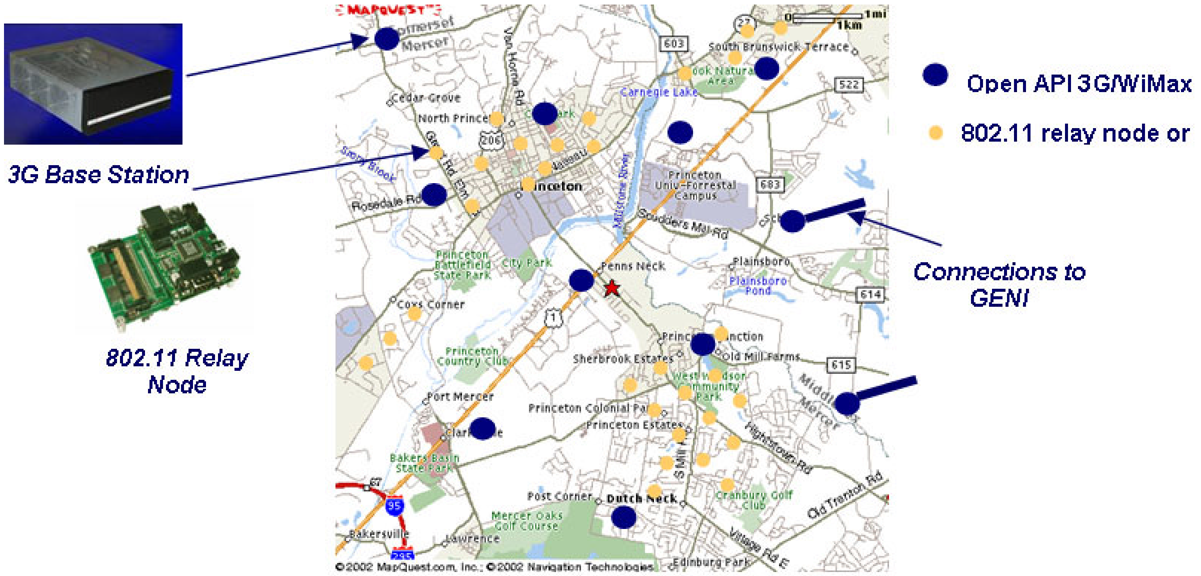


Figure 5.9: Example deployment of an open wide-area 3G/WiMax-based network across a suburban area.

GENI will include one or more wide-area wireless experimental networks with ~10 open API 3G or WiMax base station routers along with ~100 802.11 forwarding nodes and access points covering a suburban area of about 50 Sq-km. 3G nodes will need to be mounted at heights of ~30m or higher on buildings or towers, while 802.11 nodes are installed at ~8m on utility poles, etc. All radio nodes in this system will require electrical powering and a wired interface for connection to the wired GENI backbone. Mobile nodes used in experiments with 3G networks will require an open API wireless card that provides access to data link and network layer functions not available in current cellular implementations – such an open interface 3G radio module and related software will be developed as part of the proposed MREFC project. A similar open WiMax radio card / module will also be developed leveraging industrial collaborations.

Both 3G / WiMax and 802.11 nodes in the network will support flexible programming of layer 2, layer 3 and higher protocols by experimental end-users. This system uses the same dual-radio forwarding node and radio router platforms described in Sec 5.3.6, but with different 3G and WiMax radio cards as needed. Virtualization of the network can be achieved through the use of multiple radio technologies (such as 3G, WiMax and 802.11) and / or use of orthogonal frequency assignments.

5.2.8 Cognitive Radio Subnet

The third wireless subnet, a cognitive radio network, is intended as an advanced technology demonstrator with focus on building adaptive, spectrum-efficient systems with emerging programmable radios (see Figure 5.10). The emerging cognitive radio scenario is of current interest to both policy makers and technologists because of the potential for order-of-magnitude gains in spectral efficiency and network performance. NSF and industry funded R&D projects aimed at developing cognitive radio platforms are currently in progress and are expected to lead to equipment that can be used for GENI in the 2007-08 timeframe. Protocol research to be supported with the planned experimental system includes discovery and self-organization, cross-layer protocols for PHY adaptation, cooperation and competition mechanisms, spectrum etiquette procedures, and cognitive radio hardware/software performance optimization.

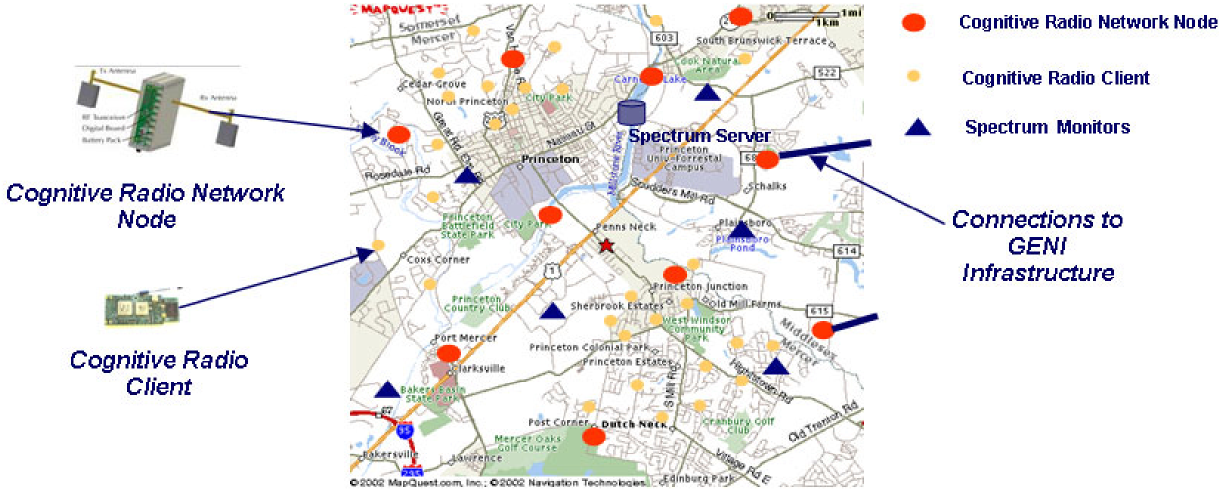


Figure 5.10: Schematic of a cognitive radio deployment in GENI.

GENI will include a cognitive radio deployment in a suburban/medium-density coverage area ~50 Sq-Km with the objective of demonstrating and evaluating this technology as an alternative to available cellular and hybrid cellular/WLAN solutions. Implementation of this system also involves construction of a distributed spectrum measurement infrastructure along with centralized spectrum coordination resources (such as spectrum broker, spectrum server). A new wideband experimental spectrum allocation will also be required to support this trial network. A total of ~50 cognitive radio routers will be deployed over the coverage area. The deployment will also include ~500 cognitive radio terminals (associated with end-user applications). End-users will be able to program their own layer 1 (radio physical layer), layer 2(data link and medium access control) and higher protocols on these devices through the experimental software interface provided by GENI.

Several cognitive radio platforms are currently under development, some with NSF funding (for example, the GNU radio from University of Utah, the programmable radio kit from University of Kansas and the network-centric cognitive radio from Rutgers University, GA Tech and Lucent). These platforms are expected to mature and be fully tested for larger scale use by early 2007, and one or more of these designs will then be converted to medium-scale production necessary to implement this experimental system. A general-purpose cognitive radio platform includes reconfigurable hardware and/or DSP for radio modem signal processing, along with one or more embedded CPU's for packet-level protocols and radio adaptation/control. Such a platform is inherently suitable for slicing, virtualization and user programming as the same piece of hardware can be used to implement multiple radio technologies that co-exist in the network at the same time. Platform software for these emerging hardware implementations will be designed to take these requirements into consideration.

5.2.9 Application-Specific Sensor Subnet

GENI will include sensor networks capable of supporting research on both protocols and applications (see Figure 5.11). Since the design of a sensor network tends to be somewhat application specific, GENI will provide necessary wireless infrastructure leveraging either urban 802.11 mesh networks or wide-area suburban wireless networks listed above, along with a "sensor deployment kit" consisting of network gateways (from sensor radios to 802.11 or cellular), sensor modules and related platform software. Research topics to be studied using experimental sensor net systems include general-purpose sensor network protocol stacks, data aggregation, power efficiency, scaling and hierarchies, information processing, platform hardware/software optimization, real-time, closed-loop sensor control applications, vehicular, smart space and other applications. Specific sensor deployments in areas such as environmental monitoring, security, traffic control, vehicular safety or smart spaces will be solicited through a proposal process leading to selection of 2-3 large-scale sensor net projects. These projects are expected to address basic sensor network architecture issues (such as service models, data integrity, data aggregation, content awareness, attribute-based dynamic binding, and low latency for closed-loop feedback) and will, in general, involve new capabilities in both wired edge and wireless access networks.

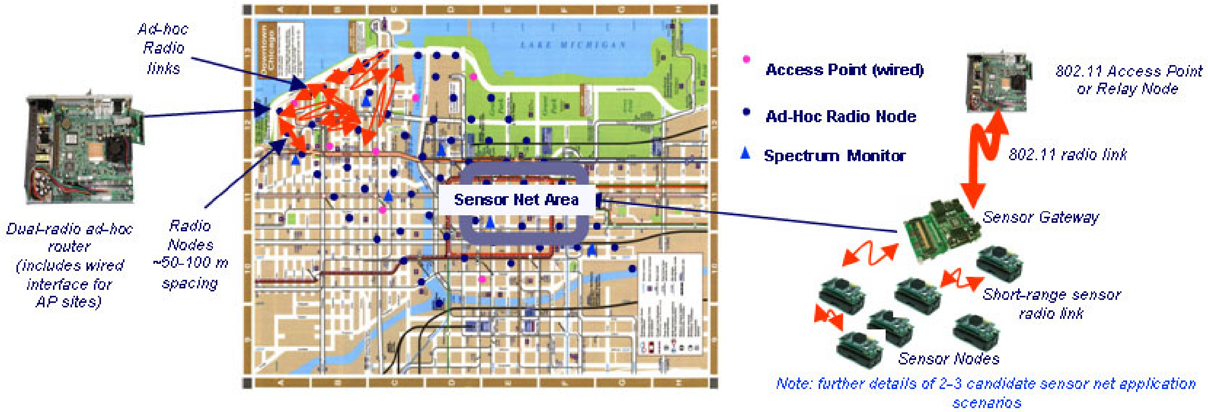


Figure 5.11: Example deployment of a sensor network.

Each sensor net application is expected to involve up to ~1000 sensors and ~100 network gateways. The low-tier sensor nodes interface with a gateway typically at distances ~10m or less, while gateways in turn connect to either 802.11 or cellular nodes with commensurate coverage areas. Platforms used in the proposed experimental system include dual-radio nodes described earlier that serve as gateways between sensor access (e.g. Mote or Zigbee radios) and 802.11 or 3G / WiMax for wireless backhaul. The deployed system will have the flexibility of migrating to newer sensor radios as they emerge, and will use commercial or semi-commercial sensor platforms (such as the MICA / Mote) consistent with large-scale experiments. Open interface versions of new sensor radios (such as IEEE802.15.4) will be developed to ensure flexibility of use. Sensor gateways will support network virtualization using multiple radio frequencies or via spatial segregation. Slicing of gateway CPU and memory resources will use techniques similar to those for programmable routers discussed earlier. Embedded sensors designed for power-efficient operation and small code size will generally not support virtualization or slicing with the understanding that multiple experiments can be supported with a new set of physical sensors (either co-located but operating at different frequencies, or at a non-overlapping geographic location).

5.2.10 Emulation Subnets

To support controlled experiments, GENI will also include emulation subnets that allow researchers to introduce artificial traffic and network conditions using both wired and wireless technology. Such emulation environments serve three main purposes. First, researchers can use emulation to develop and test their software without consuming significant network resources, such as long-haul bandwidth. After development and testing, the software can be deployed in a slice of the larger GENI infrastructure to attract real users and connect to other online services. Second, emulation provides a controlled environment where researchers can experiment with synthetic traffic, pre-specified network delays and losses, and extensive instrumentation of the system. This provides a sound way to test and evaluate the ideas. Third, emulation provides a way to experiment, in a small scale, with new technologies before widespread deployment in GENI. We envision that the emulation environment would be part of the GENI infrastructure and that researchers could run experiments that run partially on the "real" infrastructure (with real user traffic and online services) and partially on the emulated infrastructure (with synthetic traffic and delays).

We expect GENI to leverage existing successful network emulation systems. For example, the ORBIT [RAY05a] radio grid provides a 400-node experimental network similar in structure to the urban 802.11 mesh described above, and can thus be used to carry out prior protocol validation and quantitative studies. Emulab [EMU, WHI02] can similarly be used to validate complex wired + wireless protocol scenarios in advance of running an experiment on the rest of GENI. WHYNET [WHY] provides a hybrid simulation / emulation environment with a range of radio physical and MAC layers, and can thus be used to evaluate heterogeneous radio scenarios. An important design goal for integrating these emulation subnets is that of unifying existing control, management and user support software in each to a common GENI model. End-users of GENI will be able to select a suitable subset of nodes in these subnets, specify a virtualization and slicing model where applicable, and program these nodes with new protocols under consideration.

5.3 Management Framework

The management framework that overlays slices on the physical substrate is primarily implemented in software. This section identifies the three main elements of this management framework. With respect to the overview shown in Figure 5.3, the presentation is bottom up: we start with the software running on each component, then describe the core of the framework, and finish with the high-level management services running on top of the core.

5.3.1 Component Manager

Each building block of GENI runs a *component manager* that is responsible for allocating and controlling the slices embedded on that component. The component manager effectively provides a uniform *control* interface to whatever virtualized capabilities the component supports. This allows new components to be easily “plugged into” GENI. We first describe the component manager, and then return to the issue of how a component is sliced (virtualized) among multiple users.

Figure 5.12 illustrates the component manager running on a GENI node. On a PC-based node, for example, the component manager instantiates slices locally by calling the OS-provided facility to create a virtual machine. It then binds resources to the slice by making the appropriate calls to the operating system’s CPU scheduler, link scheduler, memory allocator, and disk manager. Such a component manager may also be asked to change the resources bound to a slice from time to time, as well as suspend a misbehaving slice. On a component type that does not support virtualization, but instead must allocate complete physical processing elements to slices (e.g., a customizable router that includes network processors), the component manager is responsible for managing this allocation, i.e., assigning and reclaiming processing elements.

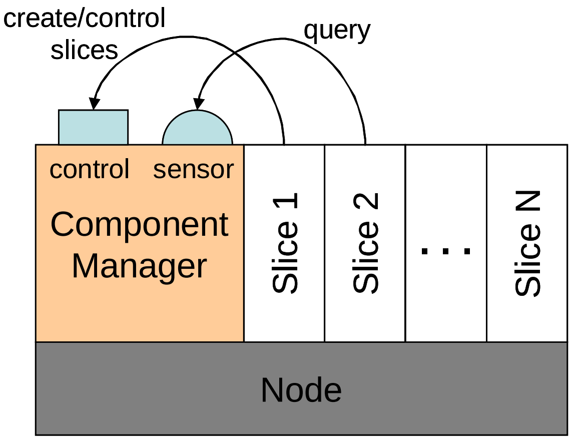


Figure 5.12: Component manager running on each building block component of GENI. The manager exports a control interface and a sensor interface. Both interfaces can be accessed by services running in a slice on that component.

A key parameter of the component manager’s control interface is an extensible *slice specification* that identifies the resources to be bound to a slice. For example, such a specification might indicate that a slice is to be allocated 100M cycles-per-second of CPU capacity, 128MB of memory, 5GB of disk storage, and 45Mbps of link bandwidth on each of three outgoing circuits. The specification can also be used to bind logical resources—such as port number, virtual circuit identifiers, and network addresses—to a slice. Since the slice specification is extensible—it is implemented as an object hierarchy—it will be able to evolve over the course of this project to accommodate the richer set of capabilities that GENI platforms will possess. The main programming task will be to port the component manager to each of these platforms, thereby linking it to the underlying virtualization methods available on that platform.

In addition to providing a means to create and control slices, the component manager also exports a *sensor* interface that reports status information about the component; e.g., its total CPU and memory utilization, the amount of CPU and link bandwidth being consumed by each slice, and so on. It is by monitoring the sensors available on each node that the global services (described below) are able to monitor the state of GENI and discover what resources are currently available; for instance, Figure 5.12 shows a slice on the node querying the sensor interface. A critical piece of this interface is an *audit sensor* that is able to map network activity (e.g., a flow of packets to some destination address at a particular point in time) to the slice responsible for that flow. This auditing sensor plays an important role in linking disruptive and potentially suspicious behavior to the responsible researchers.

It must be possible to remotely control all GENI building block components. Most of this control will be implemented by *services running in their own slice on top of* the GENI substrate (see Section 5.3.3)—these services manipulate individual building blocks by calling their component manager control interface, also illustrated in Figure 5.12. However, there is a bootstrapping problem—the substrate must provide enough networking capability to allow these services to "reach" these remote components. Initially, GENI will leverage the existing Internet for this purpose. This means that we expect the building block components to run existing control protocols (e.g., BGP, GMPLS), thereby allowing the global management software to initialize and configure the substrate. As we expand GENI to include new network technologies, and as new architectures allow us to lessen our dependency on the legacy Internet, the GENI components will need to also support the new control protocols that emerge.

Returning to the issue of how a component is sliced (shared) among multiple slices, the technical challenge is to settle on the *level* at which the substrate component is virtualized. The level of virtualization, in turn, directly impacts the kinds of experiments that can be programmed onto the component. Our approach is to give researchers virtualized access to the physical substrate at multiple levels, with different kinds of experiments enabled by each level. We identify three discrete levels—corresponding to the major classes of experiments we anticipate running on GENI—but we expect components to support much finer distinctions of programmability, according to experimental needs:

- **Overlay Level:** Allows slices to run network services and applications deployed as overlay networks on top of today's Internet. Slices have access to the full capabilities of a virtual machine, with virtual links implemented as TCP, UDP, or IP tunnels, using the conventional socket abstraction.
- **Virtual Device Level:** Allows slices to contain network architectures and services that run directly on top of layer 2 circuits. These virtual links have guaranteed performance characteristics and are accessed as virtual devices that faithfully emulate physical line cards; i.e., expose device queues, link failures, and so on.

- **Circuit Control Level:** Allows slices to run network architectures that have the ability to control—set up, tear down, and configure—circuits via multiplexing, grooming, and switching devices. Slices access this capability through a virtualized control interface.

Keep in mind that not all GENI components will support all levels of virtualization (programmability). For example, some edge nodes might support overlay-level slices, but not virtual devices or controlled circuits. Other components might support both of the first two levels, but the third level will likely be supported only in the backbone (or some subset of the backbone). Similarly, the availability of a lower level of virtualization does not mean that higher levels are absent; different researchers will chose to use different levels even on the same component. For example, researchers interested in overlay services will continue to use the overlay level even when the virtual device level is available.

5.3.2 GENI Management Core

A collection of building block components is aggregated into an autonomous unit through a *GENl Management Core* (GMC). The GMC has two primary responsibilities. The first is to instantiate slices across a collection of physical GENI components. The second is to remotely manage those components: ensuring that each component is running the right software, detecting when a component has failed and taking the necessary steps to recover, and monitoring traffic originating on GENI components so as to be able to respond to anomalies and security incidents.

The GMC can be viewed in two different ways. Abstractly, the GMC defines a universal (GENI-wide) set of agreements about various entities that participate in GENI, including users, slices, and substrate components. These agreements include object definitions, interfaces and protocols used to access these objects, and name spaces used to uniquely identify these objects across the scope of GENI. Concretely, there can be one or more implementations of the GMC that adhere to these agreements. Figure 5.13 illustrates a reference implementation, which consists of the following three modules.

- **Slice Manager:** Records the state of each slice: its slice specification, the set of components it is embedded in, and contact information for the researchers responsible for the slice's behavior. The slice manager is used to create and control slices on behalf of researchers.
- **Resource Controller:** Records information about each constituent node, link, and subnet, including the state of each resource (e.g., functioning, failed, debug), the capabilities of the resources (e.g., link bandwidth, CPU bandwidth, memory capacity, processor type), where the resource is located in the network, what software currently runs on the resource, and methods (control protocols) the GMC uses to control the remote resource. The GENI operations team interacts with the resource controller to remotely configure and manage GENI resources.
- **Auditing Archive:** Periodically uploads and archives the audit data collected on each node. This information is used for diagnostics and to resolve security incidents.

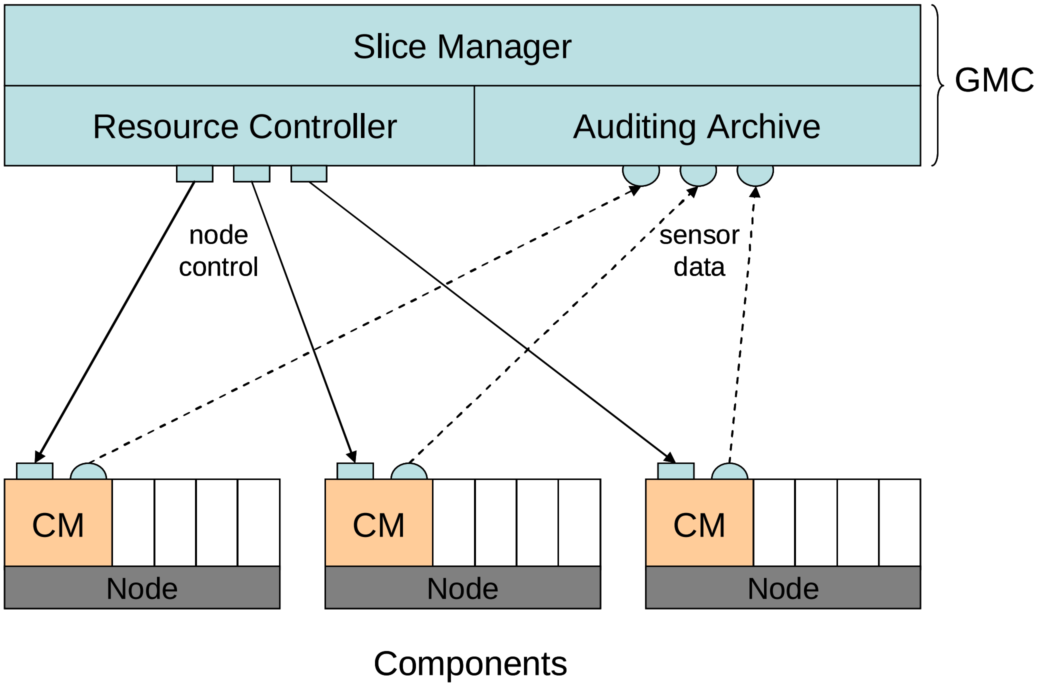


Figure 5.13: Schematic of the GENI Management Core (GMC). It includes a slice manager, resource controller, and auditing archive that interface with the individual substrate components.

There can be multiple instantiations of the GMC, each managing a subset of the components available in GENI. Allowing multiple instantiations is motivated by the desire to support decentralization. While there could be a single GMC for all of GENI, this is unlikely to be the case. Instead, we expect there to be multiple instances of GMC, each corresponding to an autonomously managed aggregate of components. We say each such GMC instance corresponds to a *management authority* that controls an independent set of GENI resources.

For example, we envision a separate GMC instance for each major piece of the substrate: the backbone, the set of edge sites, and each wireless subnet. There will also be a separate GMC for the resources contributed by autonomous organizations that *federate* with GENI. We return to the issue of federation in Section 5.4.1, but from an architectural perspective, each such organization makes its resources available through a GMC—where the architecture defines a universal naming scheme and set of interfaces that all GMC instances share—with the organization free to manage, control, and define policies for, its own set of resources.

While there will be multiple GMC instances, from the user's perspective, they can all be logically centralized through a common web interface. The user is largely unaware that GENI spans multiple autonomous systems, much as is the case in today's Internet. More to the point, however, users seldom interact directly with the GMC, but instead access GENI through a set of distributed services. These services are designed to interoperate with the set of GMC instances that comprise GENI, and run in a slice that is embedded within GENI (i.e., they have a point-of-presence on the components in the GENI substrate). We broadly classify these services as being of two types: *infrastructure* services and *underlay* services; we describe each in the next two subsections.

5.3.3 Infrastructure Services

The first set of services that define GENI's management framework, which we refer to *infrastructure services*, are used by researchers to create and manage slices. In a sense, these infrastructure services provide a ``portal" through which researchers manipulate and interact with GENI. We separate these services from the core because they will evolve over time; we expect the research community to propose new and more powerful infrastructure services as they gain experience with GENI, while the core is expected to remain relatively static. We identify four infrastructure services that GENI will provide:

- **Provisioning Service**: Used by researchers to create, initialize, and manage a slice on a set of GENI resources. The service helps researchers discover the set of available GENI resources that best match the requirements of the researcher's slice. It then contacts the component manager on each selected node or subnet to instantiate the slice on that component, and downloads the slice's configuration files and software packages onto these components.

- **Information Plane**: Used by both individual researchers and the GENI operations team to monitor the health of both nodes and the slices running on them. The service takes advantage of the sensors running on each node, as well as inter-node probes that report the state of the network and services. The information plane can be used to monitor GENI slices for anomalous behavior, as well as to log events and data for future analysis. It might also assist the provisioning service by supporting resource discovery.

- **Resource Broker**: Used by researchers to acquire and schedule GENI resources, and by the GENI governing board to impose policy on how resources are utilized. The broker must support a full range of usage models, ranging from short-term experiments to long-running services. It must also represent the incentives and interests of various stakeholders, including policy knobs that allow differential bandwidth pricing and support for different acceptable use policies among hosting sites.

- **Development Tools**: Used by researchers to develop and debug their experiments, thereby reducing their barrier-to-entry and the duplication of effort among research groups. The toolbox includes programming environments (e.g., protocol elements), as well as support for debugging, tracing, and logging. It also provides "control knobs" through with the researcher is able to steer and parameterize the experiment, and "safety envelopes" that restrict what a slice can do while it is being debugged.

These services are not intended to be completely independent, and in fact, they will leverage each other whenever possible. For example, it is likely that the provisioning service engages the information plane to determine what resources are available, and the resource broker to secure link bandwidth and compute/storage capacity on those nodes. Similarly, the development tools should work in concert with the provisioning service. In general, we expect the set of infrastructure services to form an interconnected aggregate of sub-services.

To illustrate how global infrastructure services will be deployed across GENI, Figure 5.14 shows how the Provisioning Service (PS, highlighted in yellow) is split across the GMC and the individual components. The "front-end" portion of the Provisioning Service (call it PS-GMC) watches for changes to the Slice Manager's slice state database. When a new slice is created, PS-GMC communicates, via a private protocol, with a shim (also highlighted) running on those components where the slice should be instantiated; that shim then invokes the component's control interface to create the slice and bind resources to it. The important point is that global infrastructure services themselves are distributed applications running, at least in part, in slices on the individual components in the GENI substrate.

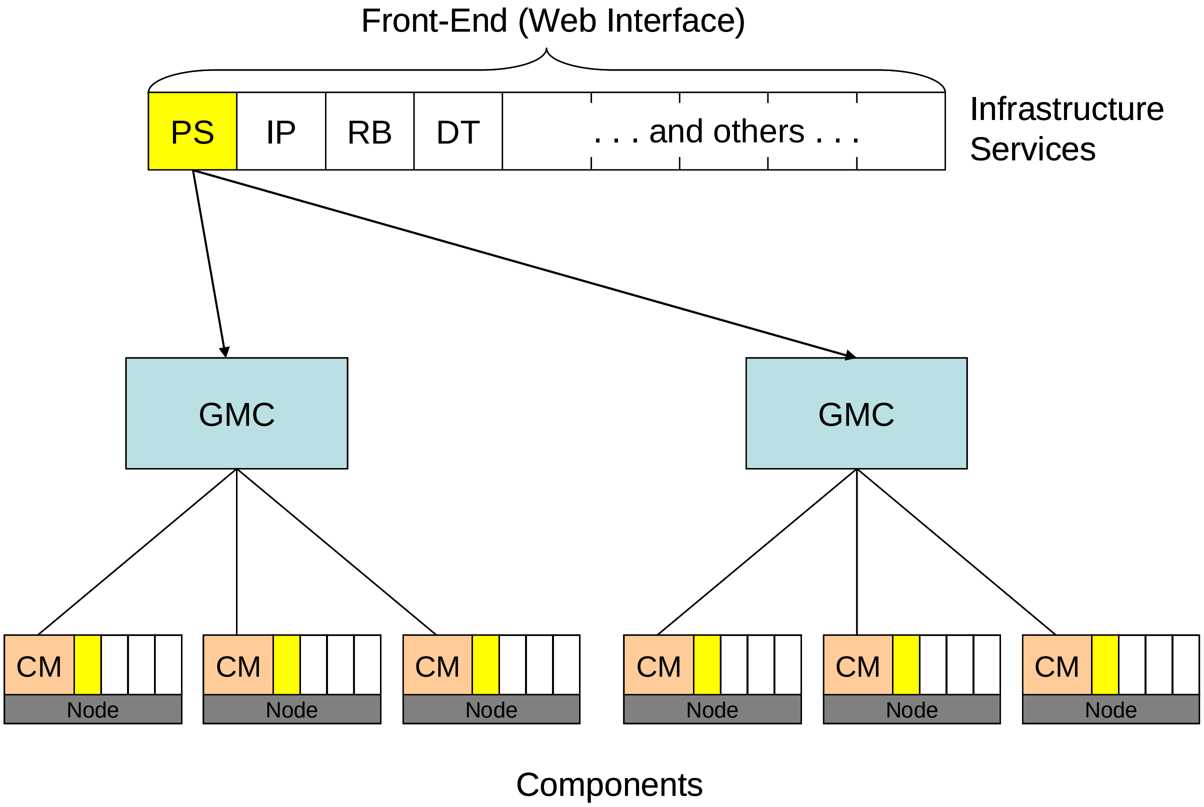


Figure 5.14: An illustration of how an infrastructure service, such as a Provisioning Service (PS) both interacts with a GMC and is distributed across a slice of GENI. Users access the set of available services through a logically centralized front-end web interface.

5.3.4 Underlay Services

We recognize that building a comprehensive network service or architecture from scratch is a daunting task. Much like an operating system that provides a set of library routines, thereby lowering the barrier-to-entry for application programmers, GENI will provide a set of *underlay services* that will be of wide value to the user community. These underlay services are similar to the infrastructure services described in the previous section in the sense that experimenters use them to help reduce their programming burden; the difference is that the former are likely called ``at runtime" to accomplish a slice-specific task, while the latter are typically used to ``manage or control" an experiment. We identify four example underlay services:

- **Security Service**: Provides a set of security-related mechanisms that equips experimental slices with the means to provide strong authentication and authorization. This set includes a public key infrastructure, delegation certificates, and mechanisms for rights management. It also includes mechanisms that can be used to detect and limit potentially faulty slice behavior.

- **Topology Service**: Provides information about what neighbors exist in the network and by what sort of links they are connected. Ideally, the topology service provides information at multiple levels of resolution; e.g., peering relationships among autonomous systems, router-level topologies, and intra-ISP topologies. The service is used by higher-level services and applications to select suitable nodes and to construct a suitable network topology.

- **File and Naming Service**: Implements a core set of distributed storage and rendezvous capabilities, enabling other services to store logs and other data to a universally accessible and persistent ``virtual disk.” The service may also include a distributed hash table that provides a lightweight and scalable naming system for distributed objects managed by an experiment.

- **Legacy Internet Service**: Implements the data and control plane of today’s Internet in a slice on top of the GENI substrate. This implementation allows GENI to bootstrap itself, and it serves as a reference implementation that can be modified to realize alternative architectures. This service includes a configurable high-speed data plane, an extensible suite of control protocols, Internet measurement tools, and proxies that simplify the task of connecting legacy clients and servers to an experimental slice.

We cannot at this early stage anticipate all the underlay services that will prove valuable, but we instead view the identification and packaging of useful underlay services as an ongoing process. Also, as with the infrastructure services, the underlay services are expected to leverage each other, as well as take advantage of infrastructure sub-services.

Building a set of underlay services has two additional benefits for GENI. First, these services serve as test cases for both the underlying physical substrate and the management core and services that support slices. It is always the case with large systems that early adopters bear the brunt of the burden of discovering what works well and what is difficult. The developers of the underlay services will play the role of early adopters. Second, underlay services are a natural result of looking for commonality among a set of competing services. It is by identifying and building underlay services that the we expect to foster the synergy needed to produce a comprehensive network architecture.

5.4 Other Design Considerations

The previous section focuses on the major software modules and interfaces that make up the GENI management framework. This section presents another perspective of the management framework by focusing on four themes that run through all the software modules.

5.4.1 Federation

It is important to recognize that GENI will not exist in isolation. It will be interconnected to the legacy Internet, which hosts assorted testbeds and other experimental facilities. It may also be interconnected with similar facilities constructed by other parties (e.g., other national governments). Thus, there is a strong need to be able to federate multiple GENI-like facilities into a truly global facility.

Toward this end, the GENI software architecture allows multiple *management authorities* to control a set of independent resources, as described in Section 5.3.2. Each such authority will run an independent GMC to control its own resources, and in doing so, is able to establish policies regarding how other communities may access its resources.

We can imagine three scenarios where federation will be important. First, there already exists a global overlay facility (PlanetLab) that will need to interoperate with GENI, thereby giving architectures, services, and applications designed on GENI global reach. Second, it is likely that other countries will build GENI-like facilities. Each can be expected to want autonomy over its own resources, but there is obvious value in allowing experimental services to span multiple such facilities. Third, we expect other communities within the U.S. to see benefit in connecting their network facilities into GENI, again for the sake of giving their users access to a wider set of resources. For example, it is easy to imagine scientific communities adding their own purpose-built sensor networks to GENI, perhaps using our sensor network subnet as a reference implementation. It should be easy to connect this sensor network into GENI. As another example, one can imagine a community with high-bandwidth needs gaining access to one or more lambdas wanting to integrate management of this capacity into GENI for the sake of taking advantage of GENI-provided services.

While we have defined the software architecture to support federation, much work remains to be done to make it a reality. In particular, we must design protocols that allow one management authority to advertise resources to other management authorities, analogous to the way BGP advertises routes in today’s Internet. Perhaps most importantly, support for federation will force researchers to address a central challenge of networking: accommodating decentralized control.

5.4.2 Security

Security considerations are integral to successful operation and widespread acceptance of the GENI facility. For this reason the need to address security matters throughout the design, construction, and deployment process is fundamental. This section outlines the motivation, requirements, and scope of GENI's security strategy. It also briefly describes the high-level architectural approach adopted to meet these requirements. Note that we focus here on the environment, supplied by GENI, that *supports* each experiment or experimental service. As a general rule, the security of an experimental architecture or service itself is part of the design of the experiment, not of the GENI facility.

Our consideration of GENI's security strategy is motivated by several factors, which together suggest the range of issues that must be addressed.

- GENI is a highly visible target. The GENI facility, because of its global scope, capabilities, and documented presence, is an inviting target for attack.

- GENI is intended to host real-world services. A key role of the GENI facility is the provision of experimental services and applications to real-world user communities. From a security standpoint, this mission is problematic because such services are at a critical point in their development. They are operationally deployed to a reasonably large user community, but are experimental in nature and immature enough not to have been subjected to the "hardening" process of a fully commercial application. For this reason it is likely that some of the experimental services deployed on the GENI facility will prove vulnerable to attack.

- The GENI facility manages shared resources. A fundamental aspect of the GENI facility is the sharing of physical resources among a large number of experiments and experimenters, some of which are potentially in competition with each other. This function introduces the possibility of critical resource depletion attacks, caused by unintentional or even malicious actions on the part of individual users of the facility.

- GENI is intended to support security-related experiments. A key motivation for the creation of the GENI facility is to support experimental research leading to the development of a more robust, more secure next-generation Internet. Of necessity, this implies that many of the experiments to be conducted using the GENI facility will revolve around testing and stressing security-related capabilities. To varying degrees, and particularly if unexpected events occur, these experiments may potentially be dangerous to other GENI users, the GENI facility, and the existing Internet

These motivations lead to a categorization of different security concerns that must be addressed within the GENI security architecture. For clarity, we enumerate these categories here.

- Security of the GENI Facility itself. The GENI facility itself—its infrastructure and control mechanisms—must be secure against a number of threats. These include attack from either inside GENI or the outside Internet, unintentional mis-configuration, and resource exhaustion caused by runaway or uncontrolled experiments.

- Security of the experimenter environment. The environment that supports each experiment, including its allocation of resources and its control mechanisms, must be secure against attacks from inside or outside the facility.

- Containment of experiments. The GENI facility must provide defense against attacks or damage to external resources caused by errant experiments. It is insufficient to rely entirely on the well-behavedness of experiments, because it is within the scope of the GENI facility to support experiments where this well-behavedness cannot be guaranteed and the potential negative consequences of non-containment are high.

- Limiting possible attacks launched from the facility. The GENI facility should provide defense against attacks or damage to external resources caused by the explicit actions of malicious users. The GENI facility, explicitly designed to deploy highly distributed services in a resource-rich environment, is technically well situated to implement DDOS and similar attacks on external resources. This category differs from that of the previous bullet in that the action is intentional.

The management framework described in the previous section enforces security at several different levels. Each building block component is expected to support isolation between slices;

this is a fundamental aspect of virtualization. Each component also exports sensor information about how its resources are being used, including an audit trail network traffic originating at that component. Several of the infrastructure services running on GENI are then involved in collecting and processing this sensor information. For example, the information plane can be queried to prove or disprove a hypothesis about resource usage and slice behavior; a logging service records auditing and other information for post-incident analysis; the provisioning service initializes a slice and establishes the resource “envelope” within which a correctly behaving experiment should run; and a resource discovery service is used to select the subset of building block components that meet the security requirements of a particular slice.

Moreover, GENI interconnection points provide the mechanism for controlled interconnection between separate GENI experiments that are allowed to interoperate, as well as between a GENI experiment and the larger Internet. Depending on requirements, a GENI Interconnection point could be implemented as a special object embedded in a separate slice, or it could be automatically embedded together with an experiment in the experiment's slice. Depending on the category and requirements of the specific experiment, each Interconnection Point might provide an appropriate subset of the following functions

- Default-permissive traffic blocking. This is a function, such as a virus filter or default-pass firewall, that allows traffic through unless some particular data pattern is detected. If such a pattern is detected, actions might include filtering the traffic, logging the traffic and the warning the researcher, warning the facility management, and/or shutting down the experiment.
- Default-restrictive traffic blocking. This is a function, such as a default-block firewall, that blocks traffic unless some particular data pattern is detected. To be sufficiently flexible for experimental use, the semantic expressivity required to describe traffic to be permitted might be quite high.
- Rate limiters and similar functions. These functions limit the aggregate characteristics of the traffic, rather than limiting its content. They are useful when the desire is to allow an experiment to proceed but bound its use of external resources.

5.4.3 Instrumentation

Being able to measure various aspects of GENI, along with the new services and architectures deployed on it, is central to this effort. To this end, instrumentation and data collection are not separate components of GENI, but are embedded throughout the software architecture. For example, each building block component is expected to export a sensor interface that reports relevant statistics about that device. Similarly, individual slices use the same interface to export performance and usage data about themselves. Multiple infrastructure services then harvest this data, both making it available to other slices for the purpose of building adaptive services, and archiving it for future analysis. In particular, we expect the information plane described in Section 5.3.3 to play this role. As outlined in the next section, a development team will coordinate this activity and ensure that data collected on GENI is effectively archived.

5.4.4 Infrastructure Renewal

Considered as an overarching artifact, the GENI research instrument is expected to have a useful life of fifteen years or more. However, it is important to recognize that few, if any, of the individual *components* of GENI will have useful lives of this magnitude. Consequently, an integral part of the GENI plan is an explicit framework for technical renewal. This section outlines the objectives and elements of that framework.

Technical renewal is important to GENI for two distinct reasons. The first is the basic need to keep the technical capabilities of the existing infrastructure and building blocks up to date. As the performance and capability of GENI's building blocks—the hardware elements that comprise the physical substrate of the facility - falls significantly behind the state of the art, the widening gap will create two problems. First, it will impose increasing restrictions on the realism of the experiments to be performed, limiting the classes of experimental research GENI can support. Second, this gap will introduce increasingly significant limitation on the ability of GENI to act as an early deployment vehicle for research results, breaking the cycle of research that GENI is designed to support. Recognition of these problems motivates the GENI designers to aim for a process of continual and incremental renewal, to ensure that the gap does not grow too large at any moment in time.

The second reason for technical renewal is more far-reaching. We anticipate that during the lifetime of the GENI facility new network-related technologies, unimagined today, may be discovered and become practical. For GENI to continue in its role as a general purpose scientific instrument for the networking and distributed systems research communities, it is highly desirable that any such technologies be incorporated into GENI at the appropriate time.

At first glance, the difficulty of maintaining this cycle of technical renewal appears quite high. Several factors contribute to this. First, GENI is *large*. If all of GENI were to be renewed at some fixed time, the cost and effort would be enormous. Second, GENI is *heterogeneous*. Renewing the technology of GENI is not simply a matter of placing an order for some large number of computers or network elements or switches. Finally, GENI, despite its size and heterogeneity, is *coherent*. All of the pieces must fit together cooperatively to accomplish its mission, and any new pieces must do so as well.

To avoid these pitfalls, we adopt for GENI an explicit policy of continual updating with two main elements. We facilitate and catalyze renewal by *lowering barriers*, and enable ongoing renewal through adequate *resources and support*. The task of lowering barriers is technical, and is primarily discussed in this section. The need for resources and support for renewal are primarily those of management and budget, and are discussed further in relevant sections.

The technical task of lowering barriers to renewal falls to the GENI design. GENI itself has an overall framework—an architecture—that determines how the individual software components and hardware building blocks that comprise GENI fit together. This architecture transcends any specific technology used within GENI, and will itself continue to serve even as specific building block implementations become obsolete and are replaced. This task, the creation of long-lived structure within which technology evolution may occur, is in fact a key goal of system architecture.

Using appropriately applied principles from computer systems science, GENI's architecture is explicitly designed with renewal in mind. Particularly, GENI's architecture allows individual components to be updated at different times and with widely differing renewal cycles, and allows new building blocks, not yet extant at the time of this writing, to be introduced into GENI with relative ease.

GENI's architecture employs three proven techniques to accomplish this objective.

- GENI is *modular*. The design of GENI is structured into components and building blocks. More particularly, to accommodate and foster renewal, the modularity boundaries within the GENI architecture are explicitly *guided* by the requirement to evolve different aspects of the facility independently. GENI's architectural modularity boundaries are intended in part to cleanly separate technological components that can be expected to be renewed at different rates, driven by different needs and developments.

- GENI's architecture makes use of *well-defined, class-based interfaces*. These interfaces, which are published specifications of the GENI design, provide clear separation between the elements of the architecture, and clear guidance for those developing or renewing building blocks. Further, the class-based nature of the interfaces lowers the barrier to creation of new or renewed building blocks by greatly simplifying the task of interfacing the building block to the overall system. Using the extensible class/subclass interfacing model, only the minimal amount of functionality that *differentiates* the new or renewed building block from other, similar components need be implemented separately.

- GENI's architecture clearly separates global and device-specific algorithms. The GENI design depends on a number of functions and services to operate correctly, and provides a number of others for the benefit of its supported users. Within the architecture, the design of these services and functions is divided into global and device specific elements, integrated through the class-based interfaces described previously. This explicit division is intended to allow both evolutionary renewal of the building blocks with minimal effort, and to support the addition of entirely new classes of building blocks during the life of the facility.

Taken together, these elements greatly lower barriers and decrease the level of resources needed to continually renew, update and evolve GENI's capabilities during the life of the facility.

5.5 Unique Capabilities

GENI is designed to meet a number of objectives central to its mission as a catalyst for networking research and early-stage deployment. This section outlines each major objective and briefly describes how it is addressed in the GENI design.

5.5.1 Sharability and Reusability

GENI's first major capability is that it supports multiple experiments, and multiple experimenters, simultaneously. GENI provides this capability through *slicing* and *virtualization*.

Slicing is the process of allocating a coherent subset (a slice) of GENI's physical resources to a specific experiment. Typically a slice will contain all of the resources needed to implement, or overlay a logical network on top of (embed a logical network within) the GENI substrate. For example, an experimenter might implement a ``virtual ISP'' by including in their slice a set of optical switches located in different geographic areas; a set of links sufficient to interconnect the switches, and a set of border routers at interconnection points between the experimental ISP and the current Internet.

Virtualization allows a single physical box to implement multiple instances of a required logical resource within the same or different slices. For example, virtualizable router hardware located at a key point in the GENI network could provide multiple experimenters with the illusion that each has access to and control over a separate router at that location in the network. Together, slicing and virtualization provide a powerful and unique capability.

Beyond basic support for multiple experiments, GENI provides support for direct, apples to apples comparison of design alternatives. This facility consists of libraries and tools that allow researchers to easily set up and repeat experiments in a well-defined environment. This environment is defined by each experiment's embedding into the GENI infrastructure together with a set of inputs (e.g., traffic data, attack scenarios, robustness challenges, etc.), and a set of outputs that include specific measurements, performance logs, and similar information.

5.5.2 Experimental Flexibility

GENI's second major capability is that it provides each experimenter with the flexibility needed to perform the desired experiment. Although a computer network is a complex system with many separate subsystems and functions, many individual experiments will focus on one or a few specific aspects of the network architecture. In practice, the vast majority of experiments will:

- Require explicit flexibility to implement and test new algorithms and protocols in the domain of the experiment. For example, a routing experiment will require the ability to implement a new routing algorithm within the experimental environment. This implies that the engine executing the routing algorithm for this particular experiment must be programmable.
- Require specific capabilities, but not necessarily the flexibility of a programmable platform, for certain aspects of the experimental environment. For example, our routing researcher may wish to reuse existing algorithms for resource management and congestion control within her experiment, but have no requirement to implement new ones herself.
- Not care about the details of certain aspects of the environment. For example, our routing researcher will certainly require that data paths exist between the routers implementing the experimental algorithm, but will often not care whether these paths are implemented by dedicated optical fibers or switched MPLS circuits.

An important corollary to this taxonomy of needs is that the experimenter benefits from *just enough flexibility*, or *just enough programmability*; anything beyond what is required for the particular experiment simply adds to the overhead imposed on the experimenter without providing a corresponding benefit. So, the objective of the GENI design is to provide each experimenter with a beneficial mix of flexibility where required and simplicity where appropriate. We note that the equation may be wildly different in different situations; a focused experiment in resource management may need programmability for only a few network functions, while an experimental deployment of an entire new network architecture may well require near-universal programmability of the supporting infrastructure.

This breadth of requirements is addressed within the embedding process that creates the environment for each experiment. The experimenter provides a description of the resources needed that fall into each of the classes—flexible, specified, unconstrained—and the embedder uses the resulting constraints to create a virtual environment suitable for the particular experiment. This is done by allocating programmable hardware where required, and by allocating fixed-function hardware (or programmable hardware preloaded with a functional configuration) where appropriate.

Note that slicing, virtualization, and component programmability are synergistic concepts that combine to create the power of GENI:

- Slicing is the fundamental mechanism by which GENI meets the simultaneous needs of multiple researchers and research communities. The process of slicing allows an experimenter to bring together disparate GENI resources (links, switches, routers, and so on) into a coherent logical configuration that meets the needs of the experiment at hand. By making the process of creating a GENI slice and embedding an experiment within the slice simple and nearly automatic, the GENI control software facilitates the use of GENI by a wide range of research users.

- Virtualization dramatically extends the scope and usefulness of the GENI hardware, and thus the reach of the GENI infrastructure. This is true for two reasons. First, virtualization allows GENI's limited hardware resources to be more effectively shared among many users. Second, and more importantly, virtualization combined with programmability allows the same hardware infrastructure to meet the needs of different experimenters and communities. This occurs when a hardware component can be virtualized at a low level, and each virtual component programmed to perform the function required within that component's slice.

- Programmability contributes to the synergy in two distinct ways. First, and in contrast to slicing and virtualization, which are focused on the *sharing* objective of GENI, programmability is central to the *flexibility* objective of the design. This is “experimenter-level'' programmability. The second role of programmability is that it frequently allows the GENI developers to implement infrastructure-level capabilities such as virtualization within off-the-shelf components. Thus, programmable devices are likely to be more easily incorporated into the GENI infrastructure and made available to researchers. This is ``infrastructure-level'' programmability.

Finally it is critical to recognize that different hardware components within the GENI substrate will have widely varying capabilities with regard to virtualization and programmability. It is the role of the slice embedder to explicitly support and accommodate this heterogeneity. For example, a routing architecture researcher interested in constructing a ``virtual ISP'' consisting of links, routers, and optical switches may find that GENI can best meet their requirements from a slice containing programmable ``routers'' implemented through virtualization, virtual ``links'' implemented as lambdas within a shared physical fiber, and actual physical optical switches. In contrast, a researcher studying optical data management might require flexibility in entirely different dimensions, and hence an entirely different slice embedding. In each case, the GENI mangement software must synthesize from GENI's heterogeneous resources the experimental environment that meets the researcher's needs.

5.5.3 Controlled Isolation and Managing Collaboration

The sections above discuss GENI's support for individual experiments. Equally important, however, is the support GENI provides at the boundaries *between* experiments. An obvious starting point would be to assume that different experiments embedded within GENI are isolated from each other. However, this is too simple. GENI must also support controlled *interconnection* of experiments. Two examples demonstrate the nature of this requirement.

- **Collaborating Virtual ISPs:** Here, a number of experimenters, perhaps studying inter-domain routing or economic collaboration models, each construct Virtual ISPs within a slice of GENI. What is required from the GENI infrastructure is essentially a transparent connection between the slices, at points specified by the experimenters.

- **DOS Attack Experiment:** Here, an experimenter concerned with resistance to DOS attacks has deployed an experimental DOS-resistant network architecture within a GENI slice. Another experimenter, acting as an adversary, has deployed a DDOS attack structure within a separate GENI slice. Here the interconnection between the slices is more limited. The first experimenter may wish to artificially restrict the attacker's range of action at different times to study specific responses, while the GENI system itself will wish to place a strict ``quarantine'' on the entire two-slice experiment to limit potential accidental damage to other experiments or the legacy Internet.

GENI provides this controlled interaction capability by supporting dynamic ``firewalls'' that can be configured and installed as inter-slice links. These firewalls are created by the GENI control software and implemented as needed on GENI hardware elements. They can be configured to provide a totally unrestricted connection, or to limit interconnection traffic to specific data types or levels. As GENI evolves, we anticipate that the range of action available to these firewalls will become richer and more adaptive.

5.5.4 Facility Extensibility

A fourth capability of GENI is extensibility of the facility over its lifetime. It is central to GENI's mission that specific technologies and classes of network hardware that *do not yet exist* be easily incorporated into GENI. This capability will allow GENI to remain useful over a much longer lifespan, support GENI's role as a low-friction vehicle for deployment of new technologies, and foster close collaboration between “device researchers” and “systems researchers”.

GENI addresses this through the design of the control software used to create slices and embed experiments. At a high level this is a constraint satisfaction problem. The GENI infrastructure consists of a large pool of resources, implemented concretely as hardware devices of different function and capabilities. The job of the GENI control software is to synthesize from this pool the slices required to support current experimenter objectives.

To support extensibility the control software adopts a two-layer *plug-in* model. In this model each resource available to the GENI infrastructure describes itself through a small software plug-in, that makes available the capabilities and attributes of the resource. These plug-ins are used to guide the operation of a general constraint satisfaction algorithm within the global GENI control software. This algorithm is used to allocate resources to slices and guide the embedding of experiments. As a result of this architecture, when new devices are added to GENI the corresponding plug-in makes available the new information needed for the global control software, which is effectively extended to understand the new device.

5.5.5 User Opt-In

Given our position that being able to evaluate a new architecture or service under real-world conditions is a critical step in the research process, it is important that mature ideas can be subjected to realistic workloads, ideally generated by live users. This has three implications on the design of GENI.

First, GENI will have wide reach. It is not sufficient for GENI resources to be located in a small number of sites, but instead, there must be GENI point-of-presence near as many users as possible. Ideally, its reach will be worldwide. Users not close to any GENI-specific resources will access services built on GENI via overlay networks. We return to the issue of global reach in section 6.8.4.

Second, for end users to take advantage of novel services or architectures, GENI supports continuously running slices. Users will not tolerate a service that runs only periodically, meaning that multiple slices supporting long-running experiments must run continuously on the shared infrastructure. Coarse-grain time-sharing of GENI resources will work for some early-stage experiments, but the expectation is that multiple mature experiments will be able to run at the same time. Virtualization satisfies this need.

Third, GENI provides mechanisms that make it easy for users to join one or more experimental networks running, and to transparently fall back to the legacy Internet whenever the experimental network cannot provide the requested service. In some cases, this will be accomplished using transparent re-direction mechanisms [PET04]. In other cases, it will require the installation of new protocol stacks in hosts and edge devices.

6 GENI Construction

This section outlines how we expect to implement the GENI design. It breaks the effort down into a set of tasks, summarizes the budget for each task, and identifies the risks involved in the effort. A description of the management organization that oversees the construction process is postponed until Section 7.

6.1 Work Breakdown

We break down the work required to build GENI into four major tasks, each of which is broken down into 2-5 additional sub-tasks. The first three major tasks involve significant development efforts. The fourth major task involves assembling the building block components into a single comprehensive infrastructure, plus on-going management of that infrastructure. One or more teams are assigned to each sub-task, providing a measure of the level of work required. For those development sub-tasks that have been assigned multiple teams, we expect the teams to operate independent of each other; they are primarily pursuing parallel or independent aspects of the corresponding task.

The development sub-tasks (corresponding to 1-12 in the following) largely correspond to the implementation of a building block component or management service that contributes to GENI. There are also facility assembly sub-tasks that defines the integration of nodes, subnets, tail circuits, and exchange points into a coherent infrastructure (corresponding to sub-tasks 13-15), and a management sub-task that includes on-going network operations (corresponding to sub-task 16). Note that there is also a 17th sub-task (team) that corresponds to the Project Management Office (PMO). A detailed description of the responsibilities of the PMO is postponed until Section 7, where we discuss the overall management of the project. This section focuses on development and assembly tasks.

- **Node Development:** Work is required to realize the three types of node technologies to be included in GENI, with each to be completed at different times during the five-year schedule. For each, the development task involves a combination of assembling and testing the base hardware components, and writing the component manager and control protocols that each node needs to support in order to "plug into" the GENI framework (as outlined in Section 5.3.1). This major task corresponds to the following three sub-tasks:
  1) **Flexible Edge Device Development** (1 development team): Deliver the hardware and software running at edge sites. The hardware will be based on clusters of commodity PCs. The software will include an OS that supports isolated virtual machines, and a component manager that allows these edge devices to be plugged into the global management framework. A prototype of this building block already exists.
  2) **Customizable Router Development** (1 development team): Deliver the hardware and software running at backbone sites. The hardware will be based on a blade server chassis with a combination of processing elements, including general-purpose processors, network processors, and FPGA-based blades. The software will include an OS that supports isolated virtual routers, and a component manager that allows these backbone nodes to be plugged into the global management framework. This sub-task starts with a configuration that uses only general-purpose processors, and so will be able to leverage the OS running on edge devices. Throughout the facility construction, support for additional processing element types will be added as it becomes available (and needed).
  3) **Optical Switch Development** (2 hardware & 1 software development teams): Deliver two independent optical switching devices, leveraging recent research demonstrations. Also deliver control software for these devices, allowing the optical switches to be controlled by a customizable router, and hence, plugged into the GENI management framework. The software development effort will prototype the control software using existing ROADM technology. One hardware design will be selected for deployment in GENI.

- **Wireless Subnet Development:** Work is required to build the five types of wireless subnets we expect to connect to GENI. For each, the development task involves selecting appropriate sites, installing access points, distributing assorted edge devices, and writing the component manager and control software that each subnet needs to support in order to "plug into" the GENI framework. This major task corresponds to the following five sub-tasks:

    4) **Urban Ad Hoc Subnet Development** (1 development team): Deliver and assemble the hardware and software running in an urban ad hoc wireless subnet. The hardware will consist of commodity processors equipped with 802.11 devices. The software will leverage the OS and control manager developed for the flexible edge devices, augmented to provide direct access to the radio devices.
    5) **Suburban Wide-Area Subnet Development** (1 development team): Deliver and assemble the hardware and software running in a suburban wide-area wireless subnet. The hardware will consist of commodity processors equipped with both 3G / WiMax radios and short-range 802.11 radios. The software will leverage the OS and control manager developed for the flexible edge devices, augmented to provide direct access to the radio devices.
    6) **Cognitive Radio Subnet Development** (1 development team): Deliver and assemble the hardware and software running in cognitive radio network. The hardware will exploit hardware that is expected to be available in 2007-08. The software will leverage the OS and control manager developed for the flexible edge devices, augmented to provide direct access to the radio devices.
    7) **Application-Specific Sensor Subnet Development** (1 development team): Deliver and assemble the hardware and software running a sensor network. The hardware will include both commodity processors serving as network gateways (these systems will leverage the OS and component manager developed for the flexible edge devices) and application-specific sensors running purpose-built software.
    8) **Emulation Subnet Development** (1 development team): Deliver and assemble the hardware and software running in an emulation subnet. The hardware will consist of commodity general-purpose processors, interconnected by high-capacity and configurable switching devices. These processors will also be equipped with commercially available radio devices. The software will leverage both the OS and component manager developed for the flexible edge devices, and adapt network configuration tools available in existing emulation facilities.

- **Management Software Development:** Work is required to develop the various software modules identified in Section 5.3.2 – 5.3.4. These include the GMC, the necessary infrastructure services, and a collection of underlay services. The software architecture is defined in such a way that each of these software systems can be developed independent of each other. Moreover, we expect each software system to continually evolve over the course of the project. This major task corresponds to the following four sub-tasks:

    9) **GENI Management Core Development** (1 development team): Deliver a working GENI Management Core (GMC), including the databases, web front-end, auditing archives, and programmable interfaces that collectively form the core of the GENI management software. This implementation effectively codifies the GENI-wide

definition of slices, users, and components, and serves as a reference implementation that can be instantiated on behalf of multiple autonomous management authorities.

10) **Infrastructure Services Development** (5 development teams): Deliver a set of distributed infrastructure services that researchers can leverage to deploy and manage their experiments. These services will include both a per-component piece (i.e., running in a slice on each node) and a front-end piece that is accessed via the web. A critical aspect of this effort will be to identify opportunities to take advantage of each other services, and to provide appropriate interfaces that make such service composition tractable.

11) **Underlay Services Development** (5 development teams): Deliver a set of distributed underlay services that researchers can incorporate into their experiments. A critical aspect of this effort will be to provide well-defined interfaces that experiments can call, thereby supporting service composition.

12) **Instrumentation, Archiving, and Analysis** (1 development team): Deliver instrumentation modules to be incorporated into the OS, component manger, and distributed services running on the GENI building blocks. It will also provide tools to collect, aggregate, and archive this data, as well as visualization and analysis tools as recommended by the research community.

- **Network Assembly and Management:** Work is required to connect the component node and subnet technologies into an end-to-end facility. This involves acquiring the necessary fiber to build a national backbone, populating each PoP of this backbone with the appropriate node types, installing PC clusters and wireless subnets at appropriate edge sites throughout the US, connecting these edge sites to backbone PoPs using the most appropriate tail circuits, and interconnecting a subset of the PoPs to the legacy Internet via the appropriate Internet Exchanges. It also involves on-going management of the resulting network. This major task corresponds to the following four sub-tasks:

13) **Backbone Assembly** (1 assembly team, integrating across sub-tasks 13-15): Assemble the national fiber facility, customizable routers, and optical switches into a coherent backbone network.

14) **Tail Circuit / Edge Site Assembly:** Deploy hardware to the edge sites and establish the necessary tail circuits that connect this hardware to the backbone and the commodity Internet.

15) **Internet Exchange Assembly:** Establish peering relationships between the GENI backbone and the commodity Internet at a set of Internet Exchange points.

16) **Ongoing Network Management** (1 management team): Provide on-going management of the facility. This will involve trouble shooting failed components, installing software upgrades on the deployed components, responding to incident reports about network traffic generated by experiments, and assisting researchers as they use the facility.

6.2 Budget

A detailed budget is included as an appendix. Here, we summarize the budget at the level of the 16 sub-tasks outlined in the previous section, plus a 17th sub-task corresponding to the overall project management office. For each, we give separate totals for capital expenses, personnel—corresponding to the teams assigned to each sub-task—bandwidth charges, and all other operating expenses.

PROJECT TASKS	CAPITAL ($K)	PERSONNEL ($K)	B'WIDTH ($K)	OTHER OPS ($K)	TOTALS ($K)
Node Development	**29,465**	**38,225**	**0**	**7,357**	**75,047**
Edge Devices (1 Team)	12,000	8,125	0	460	20,585
Cust. Router (1 Team)	4,220	8,250	0	2,064	14,534
Optical Switch (3 Teams)	13,245	21,850	0	4,833	39,928
Wireless Subnet Development	**12,003**	**33,715**	**290**	**9,508**	**55,516**
Urban Subnet (1 Team)	5,642	6,850	0	3,957	16,449
Suburban Subnet (1 Team)	1,914	7,270	20	1496	10,700
Cognitive Radio (1 Team)	3,401	7,100	0	2665	13,166
Sensor Subnet (1 Team)	446	6,145	20	640	7,251
Emulation Subnet (1 Team)	600	6,350	250	750	7,950
Management Software Development	**3,230**	**100,125**	**0**	**10,050**	**113,405**
Mgmt Core (1 Team)	325	8,125	0	850	9,300
Infra Services (5 Teams)	710	42,750	0	4,275	47,735
Underlay Services (5 Teams)	710	42,750	0	4,275	47,735
Instrumentation (1 Team)	1485	6,500	0	650	8,635
Network Assembly & Management	**1,330**	**16,920**	**44,133**	**5,895**	**68,278**
Backbone (1 Assembly Team)	380	5,665	17,376	5,700	29,121
Tail Circuits (Shared)	950	3,165	18,917	0	23,032
Internet Exchange (Shared)	0	3,165	7,840	0	11,005
Ongoing Mgmt (1 Team)	0	4,925	0	195	5,120
Project Management	**229**	**17,875**	**0**	**3,660**	**21,764**
Project Management Office	229	17,875	0	3,660	21,764
COLUMN TOTALS	**46,257**	**206,860**	**44,423**	**36,470**	**334,010**
CONTINGENCY BUDGET (10%)					**33,401**
TOTAL PROJECT BUDGET					**367,411**
PERCENT OF BUDGET (%)	14%	62%	13%	11%	100%

Note that each sub-task is composed of 1-5 teams. Each team includes a Principal Investigator, up to four development engineers, and depending upon its mission (e.g., hardware vs. software development), one or more contract engineers or technicians. Administrative support is accounted for on the basis of approximately a 10:1 ratio of staff to administrative support. To assist in the interpretation of the budget numbers, we provide the following general guidelines, which have been used in developing each of the expense categories.

- **Personnel:** Four categories of labor were selected in creating this budget: Principal Investigator, Development Engineer, Support Engineer / Technician, and Administrative Assistance. Loaded annual salaries for these categories are $350K, $250K, $150K, and $100K, respectively. These numbers reflect a mid-point between current salaries in the academic and industrial sectors. The loading assumes a 100% overhead for medical benefits, vacations, and so on.
- **Non-wage Expenses:** These expenses vary significantly from task to task, largely due to the requirements of the task for equipment that must be connected to other sites via a network. In general, hardware-based tasks are more expensive than software-based tasks. The numbers used in each line item of expense in this category were obtained as budgetary estimates from vendors or network providers, or were based on recent budgeting of these expenses in other projects. Maintenance costs were generally calculated at 15% of installed capital each year. Travel costs were estimated to be approximately $1,500 per trip, assuming that most trips involving infrastructure work require more than 2 days because these typically involve equipment installations, upgrades, and / or testing. NRE costs of installation are based on best estimates from prior installations.
- **Capital Equipment:** The cost of capital equipment is based on budgetary quotes from selected vendors, or upon knowledge of such pricing based on recent equipment acquisitions. No effort has been made at this point to incorporate "best and final" pricing numbers. Thus, we can expect the unit pricing of capital items to decrease as negotiations proceed with equipment suppliers. Some equipment, manufactured near the end of the first five-year project period, is shown as capitalized (following a development phase where equipment has been expensed).
- **Project Management:** Management costs include both staff directly employed by the project and anticipated contracted labor where special expertise is required (e.g., legal, financial).

The annualized five-year budget is summarized as follows:

TOTAL	**2009**	**2010**	**2011**	**2012**	**2013**
$367,411k	$85,460k	$69,655k	$66,372k	$74,342k	$71,582k

6.3 Work Schedule

A more detailed work flow chart is included as an appendix. Here, we summarize the key aspects of the work schedule, highlighting the dependencies among tasks / teams.

- **Component Teams:** Each component development team delivers both the hardware and the software that define the component. The hardware is delivered to the assembly team for deployment throughout GENI. The software is delivered to the global management team for installation. The software consists of two parts: a remote control module that is plugged into the GMC, and the component manager and virtualization software (e.g., OS) that runs on each component.
    - **Edge Devices:** Deployed at 100 edge sites in each of years 1 and 2, and renewed in years 4 and 5. Initial software rollout supports basic level of virtualization. Subsequent rollouts support fine-grain resource allocation, lower levels of virtualization, increased hardware heterogeneity, and optimized packet forwarding performance.
    - **Customizable Routers:** Initial configuration—server chassis populated with general-purpose processor blades—deployed to 26 backbone PoPs in year 1. Additional blades (e.g., FPGAs, network processors, and new line cards) deployed to all PoPs in years 3 and 5. Initial software rollout supports basic level of virtualization on general-purpose processors. Subsequent rollouts support fine-grain resource allocation, lower levels of virtualization, heterogeneous blade types, and optimized packet forwarding performance.
    - **Optical Switches:** Prototype optical switches based on ROADM technology deployed to 5 backbone PoPs in year 1. These prototypes are used to develop control software during years 1 through 3. Alternative switch designs are developed during years 1-3, with a winner selected and deployed to 17 PoPs in year 4. Control software integrated into GENI support for virtualization in years 4 and 5.
    - **Wireless Subnets:** Wireless subnets incrementally deployed to 5 sites during years 1 through 3. Initial software rollout supports basic level of virtualization. Subsequent rollouts support deeper levels of virtualization, fine-grain resource allocation, increased device heterogeneity, and increased control capability.
- **Management Core:** The GENI management core team builds an initial GMC framework during year 1, and plugs in controllers delivered by component development teams. The GMC framework is upgraded continuously during years 2 through 5 to take new controller capabilities into account. The management team begins using the GMC once it is available, providing continuous feedback to the global management and component development teams. Multiple GMC instances come up during years 2 and 3.
- **Network Assembly:** The assembly team deploys node components as they become available (see above), and installs network links to interconnect them. Specifically, the backbone links are installed during year 1; tail circuits are installed as devices are deployed to edge sites over years 1 and 2; and peering relationships with commodity providers established during year 1. Also, customizable routers are upgraded during years 3 and 5, and experimental optical devices are deployed in year 1 and replaced with newly developed optical devices developed by the optical teams in year 4.
- **Services:** The infrastructure and underlay services teams develop prototype services during year 1, and then roll them out during year 2, with upgrades continuously rolled out during years 3 through 5. These services both support each other—e.g., the underlay services use

the infrastructure services to create and manager their slices—and serve as early adopters of the rest of GENI, thereby providing the testing and feedback that lead to component and GMC upgrades during years 3 and 5. Annual upgrades to distributed services include:

- Support increasing component heterogeneity, federation, and workloads;
- Lower the barrier-to-entry for researchers and students to deploy their experiments; and
- Expose stand-alone sub-services that prove to be of value to end-users.

Figure 6.1 schematically depicts the high-level relationships among teams. It shows the component development teams delivering software to the management team and hardware to the assembly team. It also shows the service development teams playing the role of early adopters of GENI.

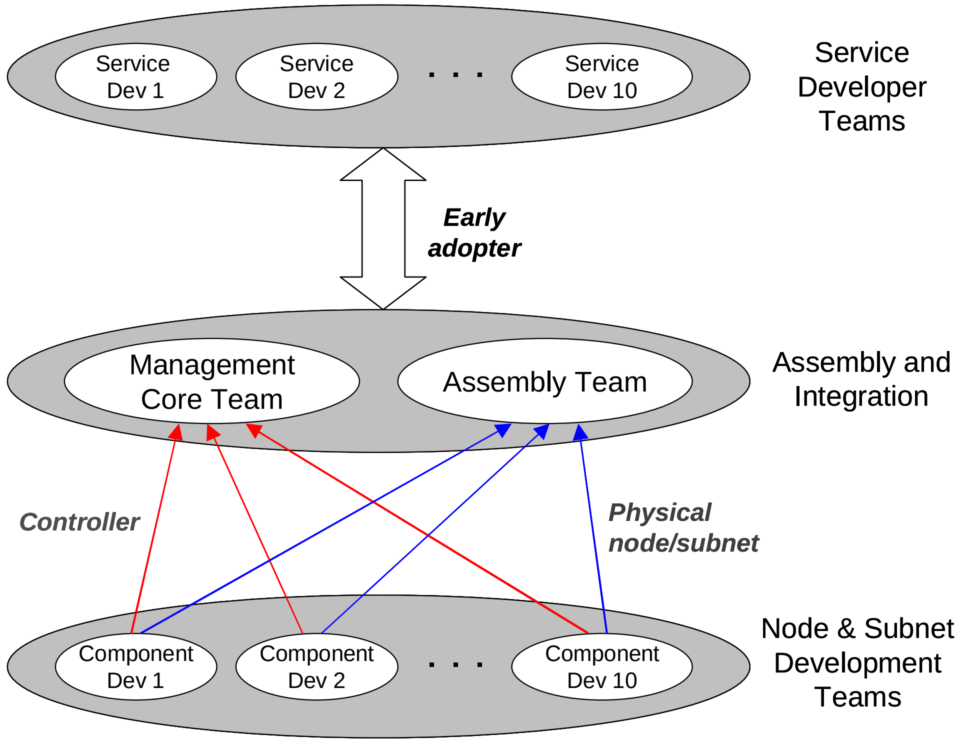


Figure 6.1: Relationship among development and assembly teams. Component development teams deliver software to be integrated into the management core and hardware to be deployed by the Assembly team. Service developers enhance GENI's functionality, but also serve as early adoptors, helping to debug and harden the substrate.

6.4 Integration, Testing, and Quality Control

We have a four-part strategy to integration, testing, and quality control. First, we have built the capability for quality control and testing into each development team. This means that resources are available for each team to test their building block component before it is integrated into GENI as a whole. Integration is primarily managed through Facility Architecture Working Group (see Section 7.2) in conjunction with the Systems Engineering Office (see Section 7.3), which defines the software architecture and interfaces into which building block modules are plugged. We encourage sound development practices that result in documentation and usage models for all software interfaces.

Second, individual building blocks, as well as GENI as a whole, are tested continuously by early adopters (corresponding to sub-task 11) along with other users. Teams are reviewed by their users, which is to say, we employ a “360 review” process.

Third, we employ a design that keeps the core GENI software as small as possible, and runs the more complex high-level services in their own slice on top of the GENI substrate. Since slices are isolated from each other, a faulty service is not able to interrupt other correctly functioning services. Moreover, it is possible to run multiple instances of a particular service at the same time, making rollback to a previous version straightforward.

Finally, GENI will employ an incremental rollout strategy for all software. This involves initially deploying new code on a small subset of alpha nodes, then on a larger set of beta nodes as it becomes stable. When code is ready for production use, it will employ an incremental rollout plan that stages the upgrade across a several day period, with the ability to rollback to the previous version should problems be encountered. In other words, there will be no “flag day” for either GENI or its individual building blocks.

6.5 Acceptance Criterion

GENI is designed to be usable by early adopters during the construction phase, and to evolve continuously as new network link and node technologies become available, and as new services are developed and refined. As such, there is unlikely to be a single discrete event that signals its completion. We do expect, however, that different classes of experiments will be enabled at different stages throughout the construction phase. Here, we summarize the major capabilities we expect GENI to provide; they are all related to the level of virtualization of the underlying hardware components. These are orthogonal from the general issues of ease-of-use and performance (which will improve in a much more continuous manner), but do serve as the major milestones for the project.

- By the end of the first year GENI will support network services and architectures deployed as overlays on top of today’s Internet. Slices will have access to the full capabilities of a virtual machine and virtual links will be IP tunnels available through the conventional socket abstraction.
- By the end of the second year slices will extend into the wireless domain. Virtualization will not be deeply embedded in the wireless subnet, but available only on wireless access points and wireless routers, again at the level of the socket abstraction.
- By the end of the third year, network architectures and services will be able to run directly on top of layer 2 circuits. These virtual links will be accessed as virtual devices that faithfully emulate physical line cards; i.e., expose device queues and link failures.
- By the end of the fourth year, virtualization will extend deeper into the wireless domain, giving slices access to virtual MAC devices. The slice abstraction will also be extended to some of the mobile devices available on a wireless subnet.

- By the end of the fifth year, slices will have the ability to control and configure multiplexing and grooming devices, including optical switches developed for GENI.

To validate each milestone, GENI will need to support multiple slices, each running a validation benchmark at a particular level of performance. It will be the responsibility of the Technical Advisory Board (defined in Section 7.2), in conjunction with the System Engineering Office (defined In section 7.3) to define the benchmark for each stage.

6.6 Transition to Operations

GENI is a unique facility designed to be continuously upgraded as new network link and node technologies become available. This means GENI will continue to evolve once the initial five-year construction phase is complete, but also that it will be ready to support researchers very early during its construction. Operational support must be built into GENI from day one.

Supporting users on a continually evolving system is a significant challenge, but one that the computer science research community has met with other systems. The networking community in particular has a long track record of building testbeds and other experimental platforms that are able to support users during their construction, and in fact benefit from early feedback about the appropriateness of the design. Networked systems are especially amenable to simultaneous use and construction because individual components can be upgraded (in either hardware or software) without affecting other parts of the system. The key is defining stable interfaces that allow new components to replace old components.

We estimate operational support for GENI **after** the construction phase is complete to be approximately $24M each year. This includes:

- The ongoing network management team identified as sub-task 16 ($1M-per-year);
- 25% of the software development costs for ongoing software maintenance and upgrades ($12M-per-year);
- 15% of capital costs for hardware that is not continually renewed ($3M-per-year);
- A fixed-budget amount set aside for bandwidth ($4M-per-year);
- A fixed-budget amount set aside for hardware renewal ($4M-per-year).

There are three things to note about these numbers. First, the operational expense **during** the construction phase is approximately $1M per year, corresponding to the work done by the ongoing management team. Second, the $8M we include for hardware renewal and bandwidth charges represent what we expect NSF and other government funding agencies to provide, and corresponds to approximately half of the expected need. We anticipate edge sites and industrial contributions will make up the difference as GENI becomes a value-added facility for end users. In fact, the $8M number is conservative in that other participants will contribute a higher percentage of the hardware and bandwidth charges as GENI evolves to be a self-sustaining facility. Third, it seems unlikely that the remaining on-going costs will fall solely on NSF as there is considerable interest in GENI’s capabilities across federal agencies and the industrial sector.

6.7 Risk Assessment and Management

It is a central tenet of business strategy that in any enterprise, there are three kinds of risks. There are risks you can afford to take—those that can be managed through creativity to be small enough probability that they do not place the endeavor in significant danger. There are risks that you cannot afford to take—the unnecessary risks that through careful planning can be eliminated. And there are risks that you cannot afford not to take—the singular occasions where the cost of inaction is simply to guarantee failure.

GENI is in the latter category. We cannot predict the future. We cannot say with a certainty that building GENI will inevitably lead to a new and better Internet. We know that GENI will enable us to understand what a Future Internet should look like, but the path from science to technology adoption is inescapably difficult to control. Nevertheless, the cost of inaction is much, much worse—an Internet increasingly marginalized because it is fundamentally insecure, fragile, and poorly suited to emerging new applications. And despite the implausibility that a few academic researchers can meaningfully influence a trillion dollar industry, history has shown exactly that—given the right tools and the right level of federal support—applied academic computer science research has repeatedly created enormous value for society [PRE05].

We believe the remaining risks are all manageable. We identify five sources: the difficulties inherent in undertaking any large scale software engineering effort; the challenges inherent in building new hardware; the possibility that the components of the system will not be integrated into a coherent system; the logistical problems of arranging tail circuits into our research universities; and the challenge of encouraging real user traffic.

6.7.1 Software

The computer science research community, funded in part by the federal government, has devoted a significant amount of its resources over the past decades to understanding how to construct large scale combined hardware and software systems, on time, on budget, and to specification. While there are numerous examples of failures by various federal agencies to successfully contract and manage the delivery of usable large-scale systems on time, by contrast the track record of the computer science research community has been stellar. In addition to the original Internet, the modernization of UNIX, production quality state of the art VLSI computer aided design tools, and production quality relational databases were all accomplished by the computer science research community using large-scale federal funding.

The characteristics of these successful large-scale project efforts provide a recipe for how to minimize the risk of project meltdown for this effort.

- Start with a well-crafted system architecture. The more complex the factorization of the system into a set of component building blocks, the greater the risk that the inter-dependencies among components will become unmanageable. The success of the Internet itself can be traced in large part to the fact that its architecture allowed components to evolve independently of each other. The GENI architecture is guided by the same design principle, whereby independent technologies can be plugged into the management framework with virtually no dependency on each other, and independent distributed services to be developed without heavy-weight coordination.

- Build only what you know how to build. Because software is plastic, there is a tendency towards feature creep; it is easier to specify the features a system "must" have, than it is to make those features work together. Left unchecked, this can result in systems that are simply too complex to work. Every major piece of the proposed software has been or is being prototyped by the research community; that is, we already know how to build it, we "only" need to make the elements work robustly together. There will be those who will complain that we are doing too little, beyond what we already understand. Our answer is, exactly, but the synthesis of these elements is revolutionary.

- Build in stages, taking the shortest distance to a working system. It is a well known result of computer science research that in software or hardware construction efforts, errors are cheapest to fix when they are caught early. We do not claim that we know enough to be able to construct error free systems; rather, we know how to catch and fix problems before they can become fatal to the overall project. The best way to do that is to put the system into active use at the earliest possible moment. We have outlined a staged construction plan, precisely to gain live experience with the system, well before final delivery.

- Leverage existing software. We expect to be able to leverage significant amounts of existing software rather than program GENI entirely from scratch. It is essential that we take advantage of such software, and just as importantly, do so in a way that allows us to also leverage the support systems already in place to keep this software up-to-date.

- Budget for mistakes. As a rule of thumb, half the cost of developing software is in finding and fixing problems; 90% of the cost of developing a successful software system occurs after it reaches its initial use. We intend to use the facility early in the construction phase so that we can correct the mistakes, or said another way, fixing the system based on experience is an integral part of the construction process.

- Design open protocols, not stovepipes. A huge point of leverage for us, versus other examples of large scale software systems construction, is that the users of the facility—the computer science research community—are themselves capable of fixing problems in the system, if we give them the right tools. This is unique to the case where we build systems for ourselves, versus building systems for other people; project meltdown is much more likely if the result is take it or leave it. We aim to build a system that continues to evolve in meaningful ways after GENI construction is complete. All of the successful examples listed above, of large scale systems being successfully delivered by the computer science research community, have the property that they continued to be modified by their user community, well after initial delivery.

6.7.2 Hardware

GENI primarily takes advantage of hardware systems that already exist or are on the verge of commercialization. In most cases, a particular piece of hardware is not on the critical path for other aspects of GENI. In other cases, we have hedged our bets by pursuing parallel approaches. We consider each major hardware component, in turn.

- The highest risk hardware development effort we plan involves building optical switching devices that will allow researchers to tap into GENI at the optical transport layer. The technology behind these devices is currently being demonstrated in the lab, but the time needed to transfer this technology into working systems is unpredictable. For this reason, the budget includes funds for two hardware development teams to work in parallel. (A third tem works on the control software for these switches.) We also note that a working optical building block component is not on the critical path for the rest of GENI.

- There is modest risk that the customizable high-speed router will not be realized with its full capabilities, but this particular device is not limited to an “all or nothing” scenario. The plan is to begin with a customizable router that employs commodity processors, making it roughly equivalent to the flexible edge devices which will be used elsewhere in GENI. New technologies can then be plugged into a shared backplane as they become available. Failure of any of these plug-ins may impact performance of the overall router, but it will not limit functionality.

- The edge devices are based on commodity processors that are readily available today. There is little or no risk that this technology will not be available in GENI.

- Of the wireless subnets, the highest risk is with cognitive radios, which are just now becoming available. The availability of these devices is not on the critical path for the rest of GENI.

6.7.3 Facility Cohesion

There is a risk that the various parts of GENI—backbone, edges, wireless subnets—will not function as a single network. This is not so much an issue of connectivity—we understand how to connect these various components at the physical level—but more an issue of whether or not the software infrastructure is sufficiently robust to allow these pieces to function as a single facility. We have already addressed the issue of software development generally, but here, we discuss how vulnerable GENI is to one or more of the 22 development teams failing or falling behind schedule.

The greatest risk is at the global level, where GENI provides a framework into which software modules for all the building block components can be plugged. This framework includes a software architecture and a set of module interfaces, which we collectively call the GENI Management Core (GMC). We are confident that we can fully define this framework by using existing prototypes as a departure point, but the architecture cannot be fully defined until we gain experience using GENI to support experimental research. Thus, the risk is that the GENI framework will be too limited to support the full diversity of building blocks, not that we will fail to define a framework that works at all. The risk to the global management framework is also managed by keeping it minimal; much of the management complexity is pushed into the independent infrastructure services.

Most of the software development sub-tasks are focused on specific building block components, and so they will proceed independent of the other teams. Failure of any one team will impact the research enabled by that building block, but will otherwise not have an impact on GENI as a whole.

6.7.4 Network Deployment

Historically, widely distributed network facilities like GENI have had trouble providing connectivity all the way to the edge. This includes both tail circuits from backbone PoPs to edge sites, as well as high-speed connectivity across campuses (i.e., from the campus IT center to computer science department machine rooms).

We believe this is no longer a significant risk. In the 15 years since NSF started trying to put high-speed circuits into universities, there has been significant build-out in campus network infrastructure. In addition, NLR, a candidate provider of fiber for the national backbone component of GENI, is actually a federation of regional providers, each of which provides edge site connectivity. We also note that GENI leverages the existing Internet, meaning that failure to connect any given location does not preclude that site from being connected to GENI; its tail circuit will simply be via a tunnel through today's Internet rather than some layer 2 circuit.

The GENI wireless subnets also face deployment risks including: (1) the availability of spectrum, particularly for the cognitive radio demonstrator; (2) potential for interference from other radio systems operating in the same region; (3) access to physical sites necessary for mounting base stations or access points; and (4) logistical issues with powering, data backhaul and remote management of radio nodes in the field.

These risks will be managed in the following manner. Regarding spectrum, GENI will use unlicensed bands such as 2.4 GHz ISM and 5.8 GHz U-NII for the majority of WiFi and WiMax radios. For 3G BTS's, local experimental licenses will be obtained from the FCC for unused UMTS frequencies, while a new experimental band will be requested for the cognitive radio demonstrator. We will also establish liaisons with major cellular operators such as Verizon Wireless and Sprint in order to manage the spectrum coordination and interference issues.

Risks related to physical site access, powering, and so on, will be addressed via cooperation with university and township IT managers in the areas of coverage. Relationships with utility companies and service operators will also be established to leverage existing infrastructure where possible.

6.7.5 Real User Traffic

One of the advantages of GENI is that it provides an opportunity for experimental applications, services, and architectures to be evaluated under realistic workloads, and more specifically, to attract real network traffic from real users. Having real users makes the evaluation of a new idea more credible, and makes the adoption and deployment path more likely. The risk is that GENI will not be able to attract real users. There are three responses.

First, we re-iterate that GENI is explicitly designed to support real users that have no affiliation with GENI. It will take advantage of overlay networks to achieve global reach (i.e., GENI is not limited to the two dozen sites on the GENI backbone), and GENI will provide client software that allows users to opt into an experimental service, yet fall back to the standard Internet when that new service fails.

Second, we do not anticipate GENI being limited to only those researchers building low-level architectures. It will also support research on distributed services and the applications that use them. Moreover, the system building research community has a history of seeking out application domains that can benefit from the distributed services they build. For example, research groups designing content distribution services have recently teamed with non-profit organizations and scientists with large files that need to be distributed to many destinations. Similarly, research groups developing multicast systems have demonstrated those systems delivering live content for professional conferences and meetings.

Third, there is evidence that this strategy works in practice. On PlanetLab, for example, content distribution networks, multicast streaming overlays, enhanced email services, large file dissemination services, and scalable naming systems run on a regular basis (24/7). These services generate up to 4TB of real traffic daily, and communicate with over one million unique IP address (not directly affiliated with PlanetLab) every day. Moreover, these numbers are conservative, as the services are restricted due to resource limitations.

6.8 Broad Participation

While we describe GENI primarily as an NSF-funded initiative, our expectation is that GENI will experience broad participation, and in fact, its design explicitly fosters such participation.

6.8.1 Academic

The academic community will play a central role in GENI. First, we expect academic and other non-profit research institutions to compete for contracts to build various pieces of GENI, as this community has a long history of contributing to major software infrastructure projects. Successful examples include the experimental network facilities that exist today (see Section 4.1), as well as the first widely distributed versions of UNIX, VLSI CAD tools, and relational databases. The Internet itself was largely built by academic and non-profit institutions.

Second, academic researchers will likely make up the majority of users of GENI. As enumerated in Section 3.3, these researchers have recently produced a wealth of innovative architectural proposals that would benefit from evaluation and deployment on GENI. Quantitatively, today there are roughly 1000 researchers using each of the PlanetLab and Emulab facilities.

6.8.2 Industrial

The computing and communications industry has a long history of involvement in networking research. For example, the experimental facilities that exist today (described in Section 4.1) resulted from joint academic/industry/government ventures that include corporate partners as diverse as Intel, Hewlett Packard, Google, AT&T, Cisco, Microsoft, Nortell Networks, Lucent, and DoCoMo. Likewise, GENI will also seek active participation from industry. This participation will take three principal forms.

First, it is expected that industry will participate in the research done on GENI in a pre-competitive setting. This is simply a continuation of the collaborative research between industry and academic researchers that has been commonplace over the last 30 years.

The second sort of participation is straightforward, based on standard contracting relationships. We envision that many elements of the GENI facility will be constructed using contracts or sub-contracts with industrial partners. Depending on the work to be performed, such contracts might be with networking equipment vendors, electronic device designers and low-volume manufacturers, bandwidth providers, and network operators (ISPs), among others. In all cases, however, this class of relationship will be marked by a clear specification of the work to be performed, with the NSF or the GENI consortium in the role of customer to the industrial partner's provider.

The third category of participation is both more open-ended and more novel. We envision the likelihood that industry partners will choose to contribute to the GENI facility for a variety of reasons not related to immediate contractual reward. These reasons might include

- More rapid deployment of technologies developed within the industry. Should GENI be successful as an early-deployment catalyst, there is no reason it should not serve this role for new industrial technologies as well as new research results.
- A more direct path for moving academic research results to commercial practice. Here again the benefit derives from GENI's role as an early deployment enabler. In this scenario, however, the deployment being enabled is that of new research results on top of industrial platforms. In essence, by designing and implementing commercial equipment that fits within the GENI model, the likelihood of new research results moving rapidly from the GENI infrastructure itself to the broader commercial setting is increased.
- Relevant industry partners will receive some incentive to contribute to GENI for a variety of "standard" reasons: e.g., public visibility and public relations.

In the design of the GENI facility and program, we wish to encourage all forms of industrial partnership and collaboration. Similarly to our requirement for continual renewal, discussed in Section 5.4.4, this encouragement takes two forms: the lowering of barriers and the creation of positive reinforcements.

Many of the steps we describe in Section 5.4.4 to lower barriers to continual renewal are also those needed to lower barriers to industrial partnership. The appropriate use of modularity, well-defined class-based interfaces, and cleanly separated global and device-specific functions combines to greatly simplify the task of integrating a vendor's products into the GENI architecture and facility. To further lower barriers to integration, we identify two further, related, requirements. These are stability of the defined interfaces, and the existence of feedback mechanisms for clarifying, enhancing, and evolving these interfaces.

Taken together, these requirements make the integration of vendor-developed components into the GENI facility as simple as possible. A well-defined, well-documented, stable architecture and interfaces that impose minimum requirements on the vendor provide the basic framework for integration. A process for transmitting understanding of these interfaces from the GENI designers to the vendor community and for providing and incorporating feedback from the vendor community to the GENI project closes the loop, lowering technical barriers to industry participation to the greatest possible extent.

6.8.3 Government Agencies

We expect significant participation from additional research communities, application domains, and the government agencies that support them. We envision this participation happening in two ways.

First, the basic ideas behind GENI have been presented to the National Coordination Office for Networking and Information Technology Research and Development (NCO / NITRD), an organization that coordinates networking R & D activities across various government agencies. The NCO / NITRD reports to the President's Office of Science and Technology Policy (OSTP). We will continue to develop GENI in consultation with this Office.

Second, we envision individual research groups and funding agencies wanting to connect their experimental facilities to GENI, thereby gaining access to new services and functionalities that are being deployed. For example, it is easy to imagine a scientific community using GENI's sensor network building block as a reference implementation for a purpose-built sensor network. It should be easy to connect this sensor network into GENI. As another example, one can imagine a community with high-bandwidth needs gaining access to one or more lambdas wanting to integrate management of this capacity into GENI for the sake of taking advantage of GENI-provided services. GENI's support for federation, as well as its ability to absorb new networking technologies, enables these sorts of activities.

6.8.4 International

GENI is designed to provide global reach, making international participation essential. While some international reach can be provided by GENI proper—e.g., by leasing co-location space around the world—our expectation is that other countries will build GENI-like facilities and we have designed GENI to support a federation of such facilities. This is already happening through international participation in PlanetLab, which includes as many sites outside the U.S. as within. Many of these sites are connected to (and sponsored by) the hosting country's national research network. This is true in Canada and throughout Europe, as well as in Japan, Brazil, India, and China.

As outlined in Section 5.3.2, GENI is explicitly designed to support federation, which gives each participating organization autonomous control of its own resources. This will allow GENI to both accommodate existing facilities, as well as expand over time as additional countries decide to participate.

7 Management

This section describes the management organization and processes that oversee the construction described in Section 6. There are two principal parts to the project management structure. The first provides technical guidance and oversight for the project, representing the needs of the research community and the capabilities of the underlying technology. The second is a prime contractor and set of sub-contractors that build portions of the facility. These two parts must be connected in the right way, so that the community is able to effectively steer the construction of the facility. We describe each part of this management structure, in turn, and explain the relationship among them.

7.1 Organizational Structure

The GENI project will be hosted by a research consortium that serves as the prime contractor. The consortium does not currently exist, but its formation is being advised by consortiums formed by similar projects. The **GENI Community Consortium** (GCC) will be a member-based organization in which scientific, educational, and research institutions will be eligible to apply for membership. The GCC will be a 501(c)(3) non-profit corporation, and as such, will have a Board of Directors with fiduciary responsibility. The GCC will have additional organizational structure, to be elaborated at the time it is created. Here, we focus on those aspects of the GCC most relevant to GENI.

The GCC will establish a standing **Executive Committee** (EC) to oversee the GENI project. The EC will consist of senior members of the academic, corporate, and government research communities, with the restriction that EC members will not compete for sub-contracts to build GENI.

The EC will have four major responsibilities. The first is to appoint people to two critical positions:

- **Project Director:** The Director is ultimately responsible for the project as a whole. This person provides technical oversight, defines and articulates the overarching vision for the project, and ensures that the various objectives of the project are satisfied. It is expected that the Project Director will be a leading computer scientist who takes a leave from his or her current position to direct the GENI project.
- **Project Manager:** The Project Manager is responsible for the overall management and execution of the project. This person monitors sub-contractor performance, ensuring that the various aspects of the project stay on schedule and budget. The Project Manager also directs vendor and carrier activities, and coordinates interactions across the scope of the project. It is expected that the Project Manager will be a full-time employee of the GCC.

Both the Project Director and Project Manager serve on the EC. The Project Director has overall authority to direct the GENI project, but serves at the discretion of the full EC. The Project Manager reports to the Project Director.

The second duty of the EC is to create a **Technical Advisory Board** (TAB), to be chaired by a senior researcher who serves as Chief Architect for the project. The TAB is described in more detail in Section 7.2.

The third duty of the EC is to create a **Project Management Office** (PMO), to be directed by the Project Manager. The PMO is described in more detail in Section 7.3.

The fourth duty of the EC is to conduct a fair and open competition through which the development teams responsible for building GENI are selected. This process involves soliciting proposals, running a set of review panels to evaluate the proposals, and finally selecting the teams to be awarded sub-contracts. (The awards will actually be executed by the PMO, which also monitors the performance of the awardees and adjust sub-contracts as conditions warrant.) The evaluation and selection process will be conducted in accordance with Federal Acquisition

Regulations. Note that because the EC will be making decisions regarding sub-contracts, it is important that EC members not be unduly conflicted by also competing for those contracts.

Once the Project Director and Project Manager have been appointed, and the sub-contractors selected, the EC continues to play an important oversight role for the GENI project, advising the Project Director with respect to broad array of stakeholders, including scientific, government regulatory, policy makers, and other governmental bodies. Technical leadership for the project will primarily be provided by the TAB, and management functions will be provided primarily by the PMO as directed by the Project Manager. As stated above, the TAB advises the Project Director and the Project Manager reports to the Project Director.

An overview of the project's organizational structure is depicted in Figure 7.1, with the development teams (sub-contractors) that actually build GENI shown across the bottom. The relationship between the management elements (e.g., TAB and PMO) and these development teams is described in Section 7.4.

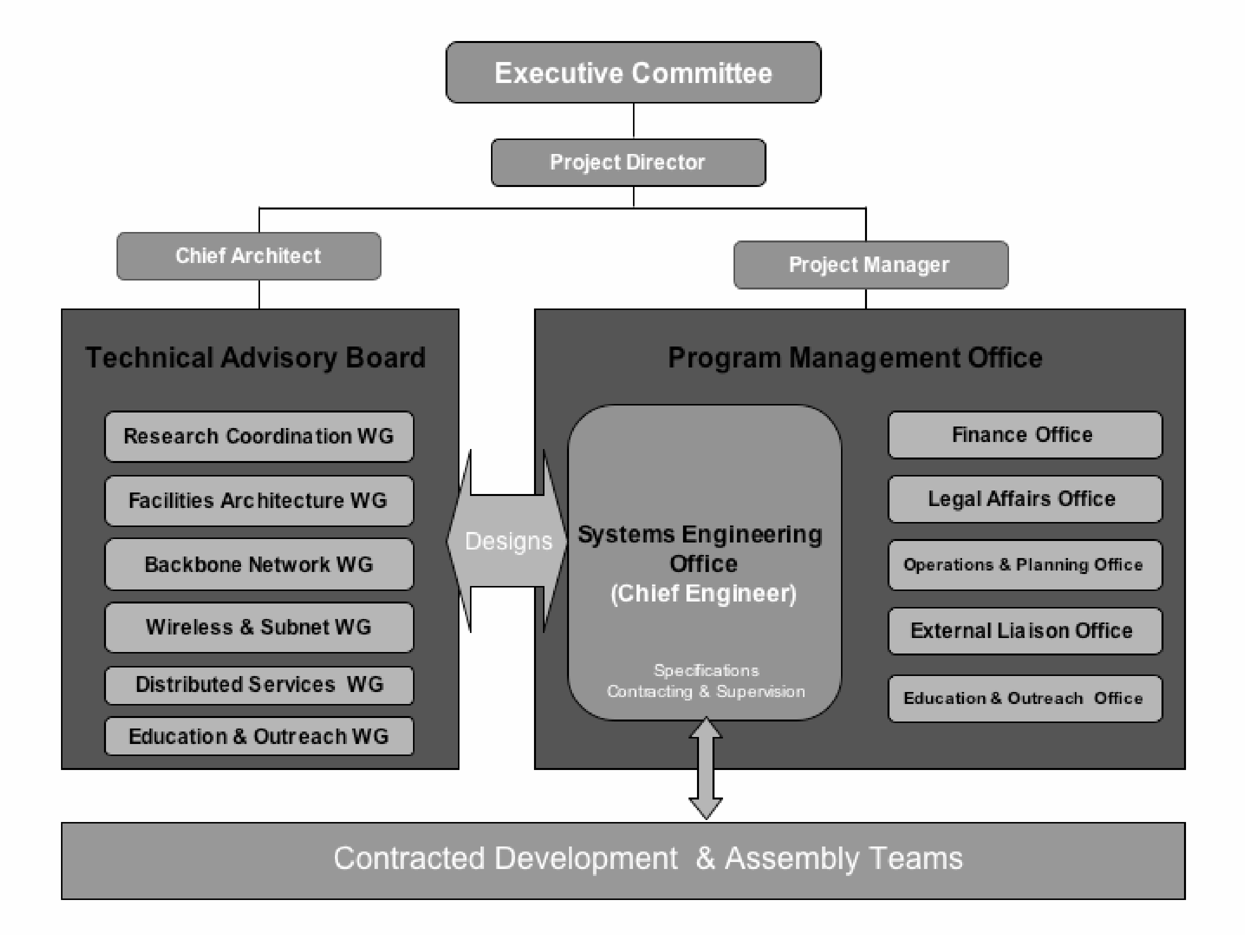


Figure 7.1: Overview of the project management structure. The Technical Advisory Board (TAB) and its Working Groups are described in Section 7.2; the Program Management Office (PMO) is described in Section 7.3; and the relationships among the TAB, PMO, and the development teams building GENI are outlined in Section 7.4. The Systems Engineering

Office is highlighted because it plays a central role in coordinating technical activity between the Working Groups and the Development Teams.

7.2 Technical Advisory Board

A **Technical Advisory Board** (TAB) provides the intellectual leadership for GENI. We expect the TAB to define polices and governance procedures for GENI, define the parameters for allocating resources to research groups, create a set of working groups focused on well-defined technical issues, and advise the Director on the construction of the facility. Members of the TAB will include the Project Director and Project Manager, the chairs of the subordinate working groups, and other senior members of the research community, as appointed by the consortium EC. The TAB will be chaired by a senior computer scientist; this person effectively serves as Chief Architect for the project.

The TAB is charged with creating working groups (and appointing a chair or co-chairs) as needed. While the set of working groups is expected to evolve—and each working group is free to organize itself around a hierarchy of sub-groups—we initially define the TAB to include the following set of working groups:

- **Research Coordination Working Group:** The research coordination working group will act as the linkage between the GENI facility and the research groups who are using GENI to demonstrate and validate their research. The group's main objectives are to coordinate use of the facility (establishing priorities as necessary), and to ensure that the requirements of the research community inform the design of the GENI facility.

- **Facility Architecture Working Group:** The facility architecture working group will oversee the definition of a logical framework that accommodates a diverse and changing set of building block components, an evolving set of distributed services, and a federation of autonomous organizations. The group's main objective is to define the overall architecture for GENI, and create the stable interfaces that enable a dynamically evolvable facility.

- **Backbone Network Working Group:** The backbone network working group will focus on the link and node technologies that will form GENI's wide-area backbone network, as well as the backbone's peering relationships with today's commercial Internet and the tail circuits that connect edge sites to the backbone. The group's main objective is to ensure that the relevant building block technologies are assembled into a working network.

- **Wireless Subnet Working Group:** The wireless subnet working group will focus on the mobile devices, RF equipment, and spectrum issues involved in constructing wireless subnets connected to GENI. The group's main objective is to guide the construction of a set of wireless subnets that enable experimentation across as broad of range of technologies as possible.

- **Distributed Services Working Group:** The distributed services working group will focus on the set of distributed services running on GENI, including a service composition framework that allows these services to leverage each other. The group's main objective is to define a set of infrastructure services that researchers use to embed their experiments in GENI, and a set of underlay services that lower the barrier-to-entry for researchers to experiment using GENI.

- **Education and Outreach Working Group:** The education and outreach working group will focus on the broader impact of the GENI facility, providing opportunities for the facility itself, along with the data collected on the facility, to be made available for education as well as research purposes. The group's main objective is to ensure that the GENI facility serves as wide of community as possible.

The main responsibility of the working groups is to produce requirement statements, design and architecture documents, and detailed interface specifications. These reports and documents serve as an evolving blueprint for GENI. The TAB then advises the Project Director regarding priorities, course corrections, and requirements based on the output of the working groups.

7.3 Project Management Office

A **Project Management Office** (PMO), headed by a Project Manager (PM), provides overall management of the GENI project. The PM and PMO is responsible for ensuring that the GENI meets the specifications of the community, as defined by the Technical Advisory Board. The PMO will develop, release, evaluate, select, and administer all sub-contracts associated with the design, development and construction of GENI. It also has responsibility for several areas related to project management, including: financial management and accounting, legal affairs and intellectual property, operations and planning, systems engineering, external liaison with government and industry, and communications and education. The PMO also convenes Advisory Panels as required to carry out some of its responsibilities. The organizational structure of the PMO is shown in Figure 7.2.

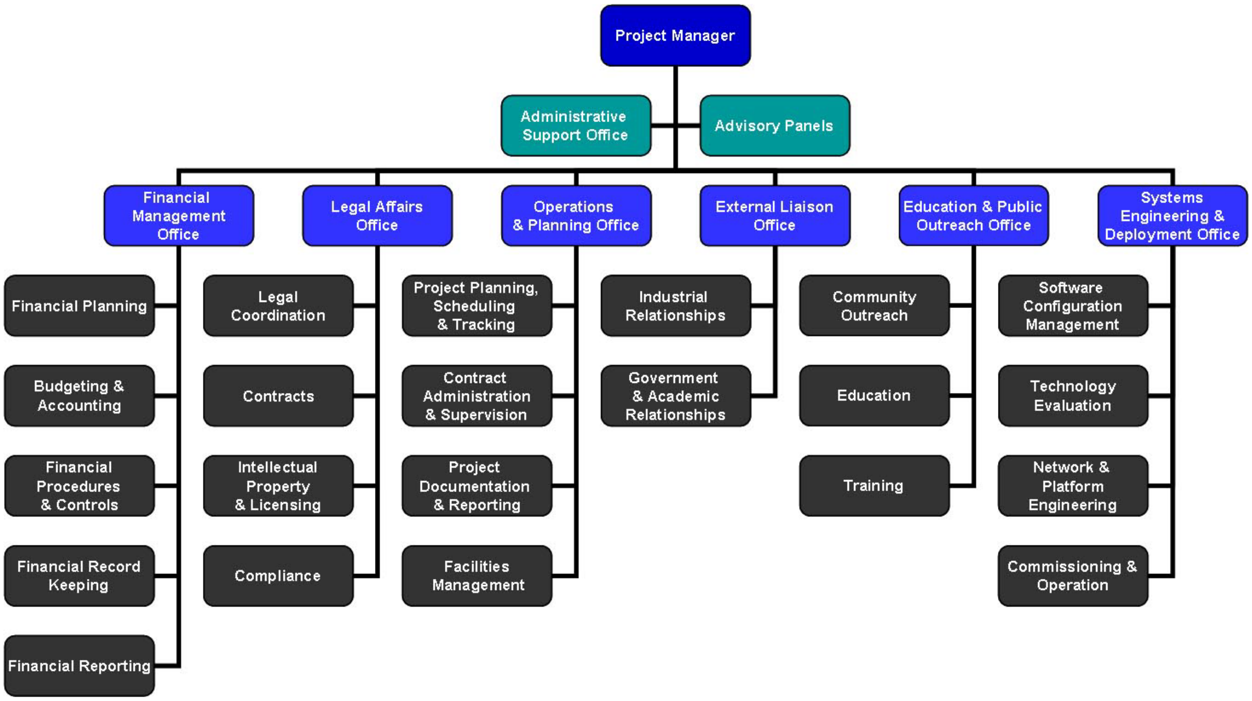


Figure 7.2: Internal structure of the Project Management Office (PMO).

The Project Manager will be a full-time employee of the project's prime contractor (i.e., the GCC) and report to the Project Director. The PM will be a member of the EC and participate in all decisions related to the content, direction, scheduling and budgeting of the project.

The Project Management Office will be organized under six principal office functions: Financial Management & Control; Legal Affairs; Operations & Planning; External Liaison; Education and Outreach; and Systems Engineering. Of these, the Systems Engineering Office is unique in its direct involvement with the teams building GENI and the working groups of the TAB. Each of these functional offices is headed by an experienced Manager with expertise specific to the functional area. Area Managers report to the Project Manager.

7.3.1 Financial Management and Control

Effective financial management and control are essential to the success of this project. Without a well-structured and well-managed financial organization, the overall success of GENI and its associated research program could be significantly diminished. To this end, the PMO will include Financial Management Office (FMO) .

The *ultimate* responsibility for financial management and control in this project will rest with the Project Director, aided by the Executive Committee (EC) and the Program Management Office. The Project Director will ensure that adequate financial procedures and controls are in place within the PMO and will, with the Executive Committee, define the framework of accountability, organizational principles, and functional relationships for management of financial responsibilities.

Key elements of the FMO function will include: (1) financial planning in accordance with the overall long- and short-term strategic plan for both research and GENI network deployment, (2) development of annual and longer range budgets that will enable timely completion of project objectives, (3) establishment of appropriate financial procedures and controls to ensure that financial objectives are met within the limits of expected project funds, (4) maintenance of records as required by good management practice, relevant laws, and regular (internal and external) audits, and (5) financial reporting. Related responsibilities include: delegation of authorities and responsibilities, adherence to standards, and holding individuals and organizations accountable for performance. Overall, the FMO must ensure that the GENI Project operates in a manner consistent with state and federal statutes, regulations, and government policies and directives. A brief expansion on some of these functions follows.

- **Financial Planning**: Overall financial planning for this project will be closely linked to the broad research and network development goals of the GENI Project. It will be essential that the financial plan ensure that these objectives can be met on schedule and within budget. This will require tight coupling of project task tracking and budget control and review. Because the Project Manager will be a member of the Executive Committee, converging scientific and financial objectives and procedures will be a regular part of EC business.
- **Budgeting & Accounting**: The 5-year budget for the development and deployment of GENI presented in Section 6 is tightly linked to the overall multi-stage strategic plan for research and GENI development, accounts for personnel, bandwidth and related network expenses, and identifies capital requirements for GENI deployment. While it is expected that elements

of planned project stages may change from time to time, the skills required to manage change and translate risk into creative research and development under a sound budget plan are normal duties of the senior academic and industrial participants expected to comprise the GENI project. For the budget to be realized annually and over the course of the entire program, the FMO will require that accounting principles and methods be established and tightly linked to other financial procedures.

- **Financial Procedures & Controls:** Procedures will be put in place in the PMO to ensure that funds received will be properly recorded and deposited according to generally accepted accounting principles. Expenditures for both capital and operating items will, likewise, be under the direction and control of the PMO, with an appropriate chain of authorization established by the Project Director and Project Manager for spending levels. Similarly, financial controls will be established to ensure that amounts owed are paid promptly. Safeguards will be put in place to guarantee this, including the assignment of authority (and backup) for all check signers. Overall financial policies will be established by the Executive Committee and become the responsibility of the Project Director to ensure that these policies are carried out.

- **Financial Record Keeping:** Financial records will be maintained for income, disbursements, and non-cash transactions in appropriate journals and ledgers that will be regularly available to the Executive Committee and the National Science Foundation and/or its authorized agents.

- **Financial Reporting:** Balance Sheets, Income Statements, and related documentation will be established and maintained to record and regularly report the financial status of the GENI Project. It will be the responsibility of the PMO to maintain such documentation and to report to the Executive Committee at regularly scheduled meetings. Similarly, Income Statements will be presented to the EC on a regular basis. An independent Audit Committee (made up of external financial experts) will be established to ensure that annual financial audits are carried out and reported.

7.3.2 Legal Affairs

During the course of this project, a multiplicity of issues will likely arise that will require expert legal advice (by the PMO to the EC and to the PMO by external experts). Of particular importance are: (1) contract preparation, either with project participants or with sub-contractors responsible for parts of the network infrastructure design, development, or deployment; (2) protection of the intellectual property of GENI, NSF and the Government, participants, and other parties (through patents and copyrights); (3) licensing of technologies, both from and to others; (4) signing of Non-Disclosure Agreements with various interested parties (e.g., vendors, carriers, ISPs, etc.); (5) dealing with inter-university legal issues; (6) international law and regulations insofar as these are reflected in the global scope of GENI and (7) ensuring that legal matters addressed by the project are properly coordinated with those of the NSF and/or other relevant government agencies.

To address these and other issues, the PMO will incorporate a legal affairs office into its overall structure. This office will include personnel with expertise in the area of government-sponsored “big science” research. Four areas are of particular interest, especially during early stages of project planning: Legal Coordination, Contracts, Intellectual Property and Licensing, and Compliance.

- **Legal Coordination:** Throughout the lifetime of GENI, the Legal Affairs Office will be required to coordinate its activities and procedures with the NSF and other government agencies involved in the project. The office will also direct legal efforts related to federation activities, particularly when foreign governments are involved. Thus, it is anticipated that the manager of the Legal Affairs Office will have prior experience in both government-sponsored research and international law, insofar as it affects legal decisions during federation decisions.
- **Contracts:** During the construction stage of GENI, multiple awards will be made to contractors to build various aspects of GENI, some for the physical network and some for large software systems development. Preparation of these contracts will be the responsibility of the Legal Affairs Office, working in collaboration with other PMO offices that will be responsible for technical aspects of the network build.
- **Intellectual Property & Licensing:** Care must be taken that liabilities related to the unlawful use of others intellectual property is avoided. In the same way, intellectual property accrued through research on GENI must be protected through licensing agreements.
- **Compliance:** It will be the responsibility of the Legal Affairs Office to ensure that GENI is operated within the legal boundaries established by the federal government. All methods and procedures adopted for the management of GENI will be subject to legal review related to government requirements, including financial management, project reporting, inventory, and internal and external auditing.

7.3.3 Operations and Planning

Several activities within the area of operations and planning that will require the regular management and leadership of the PMO. These include: (1) project planning, scheduling, and tracking; (2) contract negotiation, issuance of RFIs, RFQs, RFPs, and contract supervision; (3) project reporting, including reviews, presentations, papers, required reports to NSF and other government organizations, press releases, and others; and (4) facilities management (i.e., at GENI installations in PoPs). For this reason, an Office of Operations & Planning will be created within the PMO. This Office will be managed by someone with significant expertise in multi-dimensional, government-sponsored research or development and construction projects.

- **Project Planning, Scheduling, and Tracking:** A principal role of the PMO is project administration, including planning, scheduling of tasks, and the tracking of the status of these tasks as they are carried out by the Development Teams or sub-contractors to them. The PMO will ensure that project tasks are properly scheduled (within time, staff, and budget constraints), that schedules are maintained and milestones met, and that appropriate responses to scheduling issues (e.g., delays in one or more elements of the project) are addressed in a timely and economic manner in order to mitigate risk within the overall project.

- **Contract Administration & Supervision:** All contracts let during the course of this project will be administered from the PMO under the guidance of the O&P Office (drafting of contracts will be the responsibility of the Legal Affairs Office). Following prioritization and authorization by the Technical Advisory Board and Project Director, the PMO will be given the authority to proceed with contracts to execute specific project jobs. When contract awards are made, the PMO will monitor contractor performance and costs to ensure on-time, within specification, and within budget deliveries. The O&P Office will coordinate closely with the Finance Office and the Legal Affairs Offices in the matter of contract administration.

- **Project Documentation & Reporting:** Documentation of project results and progress are also important functions of the PMO. This task will be under the direction of the O&P Office, which will produce regular reports on the status of the project through regular reports to the NSF, and through research papers, conference presentations, proceedings, recordings, and press releases. The O&P Office will also have the responsibility for permanently archiving this documentation so that it can be made available to future network users.

- **Facilities Management:** Finally, the O&P Office will have responsibility for overseeing the preparation of facilities that will house GENI equipment and, where appropriate, for the management of these facilities in order to ensure their safe and efficient operation. This work will include site inspections prior to any installation, specification of installation requirements in accordance with good engineering practice and with the law, and adherence to local, state, and federal requirements. The Facilities Management function will supervise all installations in the network and conduct periodic inspections to maintain safe and effective operations. This office will also be responsible for issues related to the environmental impact of GENI facilities and, in collaboration with the Legal Affairs Office, prepare reports on environmental and safety issues as required. The Facilities Management function will have the responsibility to the Finance Office to provide accurate budget information for annual new installations and upgrades or maintenance to existing installations. This organization will be the principal day-to-day interface to carriers, service providers, or other organizations whose facilities GENI might use.

7.3.4 External Liaison

In the course of its operations, GENI will establish relationships with a multitude of organizations and individuals who will bring to it specific areas of expertise, special facilities or equipment, and provide advice on the direction of GENI's research and development programs. These organizations will come in the form of funding agencies (e.g., NSF and others), advisory committees, industrial participants, third-party services developers, and others. Proper coordination of activities with such organizations will be essential to GENI's success and the overall view of GENI from the larger community of scientists and the public. Therefore, the PMO will establish as one of its functional areas the GENI External Liaison Office. This Office will have the responsibility not only of coordinating the relationships with external groups and individuals, but also of *initiating* contacts and developing relationships with those organizations and people who can aid GENI's progress. Some areas of particular importance include the following:

- **Industrial Relationships:** Feedback from the industrial community will be important to the ultimate success of GENI. Therefore, even during the further planning stages of GENI, the PMO intends to establish relationships with selected industries through the formation of Advisory Panels that can be drawn upon both during the planning stages and as construction of GENI progresses. These panels of industrial experts—both in technical and in business areas—can be expected to assist in the guidance of GENI deployment. Areas of particular importance are: network design (including connections to legacy networks) new technology deployment, and even potential services and applications that GENI might have to support in the future.

- **Government & Academic Relationships:** It will also be important for the External Liaison Office to maintain regular communications with government, independent research, and academic institutions – particularly those that can provide advice on the management of large science projects near the scope of that proposed for GENI. We will begin to identify these institutions even during the early planning stages, and expect to maintain close relationships with them throughout construction and into the continuing operations of GENI.

Liaison will also have to be maintained with standards groups, such as the IETF, so that research results produced by GENI scientists and/or developers will impact the adoption of standards for the future Internet. Finally, since GENI is intended to be an international (i.e., global) network, liaison with “off-shore” networks in Europe, Asia, South America, and others parts of the world, will be critical to spreading the benefits of GENI. These, and other liaisons, will be developed and maintained under the External Liaison Office.

7.3.5 Education and Community Outreach

An important property of GENI is that it lowers the barrier-to-entry for researchers wanting to evaluate new network architectures, services, and applications. This property applies equally well to teachers wanting to give their students experience with architectures, services, and applications running under realistic network conditions: students and class projects can run in their own slice of GENI. Because GENI enforces isolation among slices, such efforts will not interfere with other research projects, and vice versa.

Moreover, unlike a centralized facility, GENI is by its very nature distributed over as many sites as possible. It is important that GENI has diverse points-of-presence—as oppose to being limited to a small number of backbone nodes—so as to allow a wide range of end users easy access to the services it provides. This is primarily accomplished by not limiting GENI just to sites that must be connected by high-speed tail circuits. In fact, we expect the majority of edge sites to be connected to GENI via today’s Internet, with particular attention paid to getting nodes into minority institutions, liberal arts colleges, and EPSCoR states.

We plan to educate the community as to GENI’s capabilities through several means. In order to do this, we will establish an Education & Community Outreach Office in the PMO. The Education & Community Outreach Office will be responsible for the *implementation* of programs that communicate with the broad community and that provide education and training related to GENI. This community includes the general public, potential GENI research users, industry, academia, and the government. This Office will work closely with the Education and Outreach

Working Group to ensure that the broad goals and objectives of GENI are communications to the widest possible audience. Policy matters related to education and training will remain with the Education & Outreach WG; execution of those policies will be the domain of the Education & Community Outreach Office within the PMO.

The full implementation of the Education & Outreach Office will have three principal roles: (1) communication to and interaction with the broad community on the work of GENI, (2) support for the development of education at the undergraduate or graduate levels in networking, services, applications, and technologies as related to future networks, and (3) development and delivery of technical training to GENI users. Work in each of these areas will begin during the next planning stages for GENI and then continue into the construction and operations stages.

- **Community Outreach**: One of the main elements of the community outreach component of this effort will involve the continuous update of the GENI website. This effort, already started during the conceptual stage of planning, is expected to be one of the best ways to get the GENI message across to a broad audience and to be a source of information and feedback to the GENI planning effort throughout the various planning stages. Community outreach will also occur by means of presentations made at regular technical conferences as well as to conferences of a more general nature, including trade shows that are often populated by commercial communications industry technical staff and vendors who will provide the future platforms for network deployment.
- **Education:** Education is expected to be a significant part of the GENI project. After all, the long-term users of GENI are today’s students. Thus, the education element of the GENI project will contribute to university education through lectures in university classes, development of educational materials, and training of students to use the GENI network in research projects for graduate degrees.
- **Training**: While GENI is in the construction stage, research will actually begin. There will be a need for training materials and tutorial sessions for the GENI users. This effort will be enabled by software services being developed to run on GENI, but additional materials and tutorials will also be required, especially during the early construction stage when GENI user resources are still under development. It may be possible to use the GENI website for some of the early training and the opportunity will explored during planning periods prior to construction.

7.3.6 Systems Engineering and Deployment

The Systems Engineering and Deployment Office has the principal responsibility for overseeing the design and deployment of GENI. Of all the PMO offices, the Systems Engineering Office is unique in that it works closely with the TAB and its working groups—serving as their “operational arm” —as elements of GENI are defined and deployed. The area manager for this office, therefore, effectively serves as the Chief Engineer for GENI.

The Engineering Office is expected to carry out several specific functions, including: (1) the development of technical specifications required to implement the GENI network services that support users; (2) the actual engineering of the network, including interfaces to all network elements, links between network nodes, intra-office connectivity, and connections to edge sites;

(3) the testing, evaluation and selection of network platforms, including those built under the GENI project; (4) planning for the deployment of the network in collaboration with sub-contractors, and finally (5) management of all design efforts in terms of configuration control and documentation.

- **Software Configuration Management:** GENI requires a substantial software development effort, involving multiple teams and sub-contractors. To meet project goals on schedule, it is essential that a well-defined *software configuration management plan* be in place *before* the start of the construction phase. Fortunately, software management is well understood. It ensures that changes to software are made in an orderly, controlled, and well-documented way. The plan involves configuration identification, base-lining, configuration control, configuration management status accounting, interface control, sub-contractor control, and software configuration audits. All of these components will be implemented in a software management process that will ensure that the software development envisioned for this project occurs in a manner consistent with good development practice.
- **Technology Evaluation:** For GENI, technology evaluation is a very broad area, impacting both software and hardware designs, as well as many areas of network design, including transport, switching, routing, signaling, control, and management. Ultimately, this function within the Engineering Office will develop the technical specifications required for RFIs, RFPs, and RFQs. It works in collaboration with other Offices within the PMO to release these documents, evaluate responses, and recommend selected vendors / contractors to the PM / PD, as well as to the TAB and EC.
- **Network & Platform Engineering:** This function develops the final technical specifications for platforms, links, and all other devices and / or systems that become a part of the GENI facility. It also develops the detailed network design for GENI. In this activity, the Engineering staff augments the appropriate Working Groups, and works with other parts of the PMO to acquire service and performance requirements.
- **Commissioning & Operations:** The PMO will play a significant role in the operational management of GENI, especially as construction nears completion. This function of the Systems Engineering Office will supervise and coordinate development contractors and the management team; develop acceptance criteria and procedures for the facility; and oversee the commissioning and turnover of the final operational facility.

Across all these functions, a critical role of the Systems Engineering Office is to act as the primary technical contact point for sub-contractors, thereby avoiding a situation in which a development team has to answer to multiple working groups. That is, this office is the official “keeper” of the technical documentation produced by the working groups, which in turn guides its supervision of the development and assembly efforts undertaken by the sub-contractors.

7.4 Workflow Management

Figure 7.3 illustrates the cycle of workflow involved in the construction of GENI. The diagram shows the flow of ideas and requirements from the research community using GENI, along with the flow of expertise of the development teams building GENI, converging on the set of Working Groups. This flow is in the form of people, with members of the research community and the development teams jointly participating in the Working Groups. These Working Groups are created by the Technical Advisory Board, with a member of the TAB serving as chair of each Working Group. The working groups, in turn, produce documents that inform the TAB, and the TAB instructs the PMO to execute the construction plan accordingly. Finally, the PMO awards contracts that result in GENI's construction.

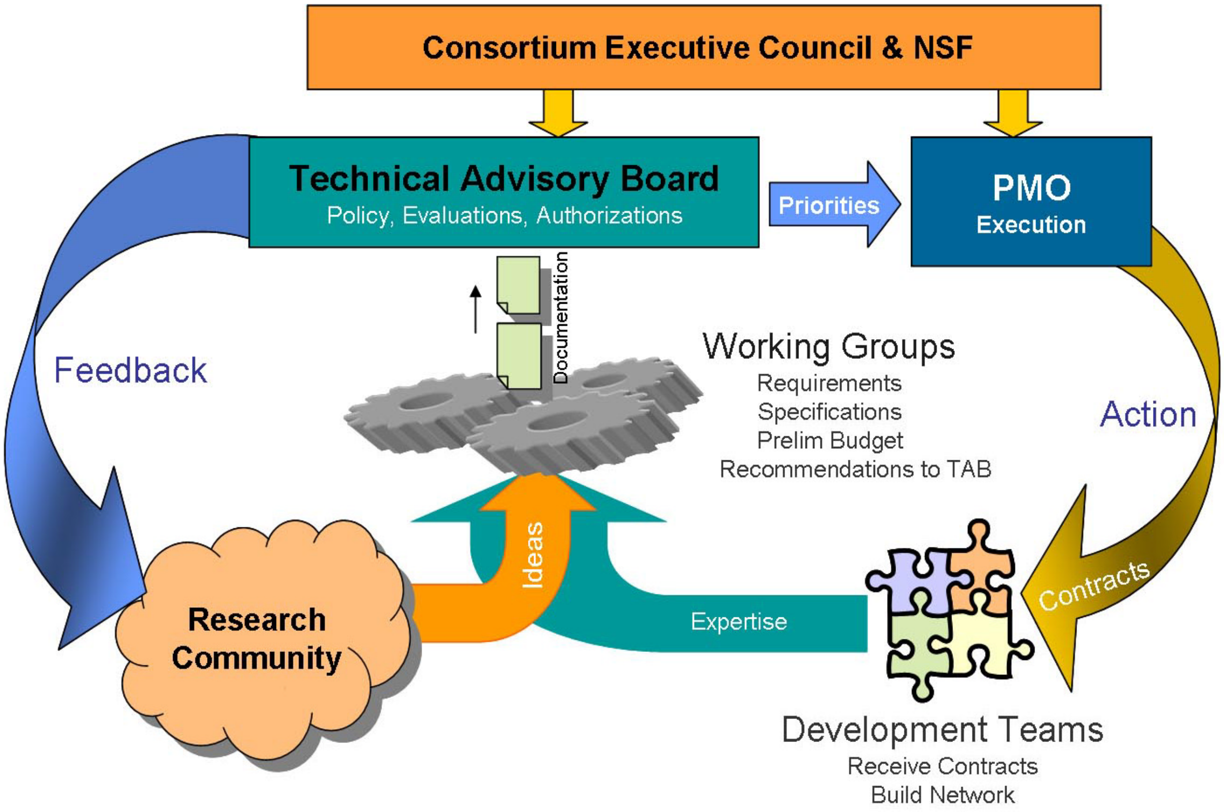


Figure 7.3: Workflow for GENI construction. Researchers and developers participate in Working Groups, which generate design documents and interface specifications for consideration by the Technical Advisory Board (TAB) The TAB defines priorities, which are implemented by the Program Management Office (PMO). The PMO executes the plan by directing sub-contractors (i.e., development and assembly teams).

Responsibilities and interactions will be as follows:

- The working groups' primary responsibility is to write requirement statements, design and architecture documents, and detailed interface specifications. Members of both the research community that uses GENI and the development teams building GENI serve on the working groups. Although not shown in the figure, the Systems Engineering Office of the PMO provides operational assistance to the working groups in this process. An initial set of working groups, and their mission statements, is given in Section 7.2.
- The TAB establishes working groups, as needed, to address technical issues involved in the construction of GENI. Informed by the requirement and specification documents produced by these working groups, the TAB then advises the Project Director in the establishment of

priorities and the issuing of work tasks to the PMO for action. The TAB also announces its priorities, policies, and plans to the research community, thereby keeping it informed as to GENI's development. The role of the TAB is defined Section 7.2.

- The PMO executes the construction plan as directed by the TAB. This involves assigning the System Engineering Office to work with the TAB to flesh out design specifications; directing the sub-contractors (i.e., development teams) to implement the specifications; monitoring that the sub-contractors are delivering according to specifications; and initiating new sub-contracts as appropriate. The structure of the PMO is given in Section 7.3.

- The teams develop the GENI components, assemble the components into an interoperable network, and manage the use of the facility, as instructed by the PMO through a set of sub-contracts. The relationship among the development teams is spelled out in Sections 6.1 and 6.3, and schematically depicted in Figure 6.1. Individual members of the development teams participate in the working groups, bringing their expertise about the underlying technology to the discussion.

- NSF and the Executive Committee of the GENI Consortium provide oversight to the process. They appoint the Project Director and approve the TAB members; they appoint he Program Manager that heads the PMO; and they select the development teams that are awarded sub-contracts. NSF, specifically, interacts with the various offices of the PMO as spelled out in Section 7.3.

Looking across the set of working groups, development teams, and management offices involved in the workflow, we identify the critical subset that is primarily responsible for integrating the various components into a coherent facility. They are:

- The Facility Architecture Working Group is responsible for defining the architecture (logical framework) into which the various pieces of GENI can be plugged. The GENI Management Core (GMC) is the centerpiece of this framework. The Chair of the TAB is expected to also serve as Chair of this working group.

- The Systems Engineering Office of the PMO is the operational arm of the TAB, including the Facility Architecture Working Group. It assists the working group in preparing comprehensive interface specifications, and is responsible for direct interactions with the various development and assembly teams. The manager of this office effectively serves as the Chief Engineer for the project.

- The Management Core team is primarily responsible for integrating the software components produced by the other development teams into the GMC, and providing a coherent "front-end" for GENI. Researchers access GENI through this front-end, and the management team uses this front-end to provide operational support.

- The Network Assembly team is primarily responsible for integrating the hardware components produced by the other development teams into an interconnected physical substrate.

Note that working groups other than the Facility Architecture group produce specification documents (also assisted by the Systems Engineering Office of the PMO) and these documents contribute to GENI's definition. However, responsibility for these aspects of the facility is effectively delegated to the various working groups according to the logical framework produced by the Facilities Architecture working group. It is this core architecture that defines how these various pieces fit together to form a coherent whole.

In summary, the intent of this method of workflow management is to ensure that the best ideas and designs are realized as GENI is developed, and that all of the relevant voices of the GENI community are heard before ideas are permanently fixed in new hardware and software installations. The effect is to provide on-going adaptability and a process for change control management. However, responsibility for effecting these changes rests with the Project Director and the Project Manager.

7.5 Operational Management

The process described in the previous section is focused on the construction of GENI. There is also ongoing operational management of the facility. This responsibility primarily falls to the management team described in Section 6.1 (sub-task 16). However, the Technical Advisory Board plays a significant role in this process. Its responsibilities include: (1) establishing policies governing how GENI resources are shared among slices; (2) establishing policies regarding what experiments are appropriate for GENI, especially with respect to their potential security impact; and (3) overseeing the federation of GENI with other network facilities.

Of these, the second—overseeing the appropriateness of experiments from a security perspective—is perhaps the most critical. As mentioned in Section 5.4.2, we will need to evaluate the potential security impact of experiments to be deployed within GENI. The first and foremost consideration is that any such assessment be seen by the research community as necessary, reasonable, minimally intrusive, and as simple as possible to perform. Establishing a procedure to make these assessments falls to the TAB as a whole, with the Research Coordination Working Group playing a key role by making recommendations. This will likely involve appointing a security *assessor* panel to consider specific cases.

In designing this assessment procedure, it is important to consider some inherent limitations in the assessment process: (1) the assessor is not in a position to verify the experiment characterization described by the researcher; (2) there is no assured protection provided in the event that the researcher is sloppy, honestly mistaken in his or her assessment, or even malicious; (3) an "experiment" is in fact an allocation of resources for a fairly uncontrolled, wide range of activities, where changes in the experiment or adjustments by the experimenter may invalidate the assessment without knowledge of the assessor; and (4) the assessor cannot insure that operational procedures for an experiment are properly followed.

These significant limitations in the role of the security assessor suggest that the assessment process cannot be expected to be accurate or complete. Rather, it is best viewed as a check and balance on the researcher, to whom the primary responsibility for running a safe experiment falls. Given these limitations, a possible set of guidelines for security assessment might include:

- Be somewhat conservative, particularly when experiments appear risky or request specific sorts of resources.

- Expect mistakes, and consider their possible effects.
- Use judgment about the previous experience and track record of the researcher.
- For higher-risk experiments, encourage or require the researcher to state bounds and qualitative behavior characteristics that can be verified by the GENI monitoring infrastructure during the experiment. Then, actively monitor the experiment. Detection of an experiment that exceeds its stated characteristics may, depending on the situation, limit the behavior or shut down the experiment in real time, and trigger a re-evaluation of the experiment between the researcher and the assessor.

7.6 Contingency & Change Management

No matter how well a project has been planned, there are situations that dictate a change of direction is necessary, and unexpected circumstances that must be taken into account. We want to ensure that such changes and adjustments can be taken in a timely manner and that valuable research funds are not jeopardized by such actions. There are two overriding aspects to this process.

First, it is important to recognize that basic management structure of GENI is explicitly designed to facilitate continual change. This is due in large part to the nature of an effort dominated by software development. As described in Section 7.4, the TAB and its working groups have the responsibility of advising the Project Director in setting and correcting the course for the construction of GENI. The PMO then executes these course corrections through its sub-contracts with the development teams, working closely with the Systems Engineering Office to adjust specifications accordingly. The development teams will need to be selected, in part, according to their ability to adapt.

Second, we have set aside 10% of our proposed budget for the effective management of contingencies. This contingency budget will be controlled and administered by the Project Director, as advised by both the Executive Committee and the Technical Advisory Board. We also note that the multiplicity of development teams provides significant flexibility in how unexpected work tasks are assigned, above and beyond the allocated contingency budget. The challenge of GENI is not how to enable change, but rather, how to constrain change. The Project Director and TAB must establish very clear milestones and timelines, and the Project Manager must enforce them.

8 Concluding Remarks

The rationale for GENI laid out in the opening sections of the document is compelling: if the Internet is going to deliver increasing value to society, then we must invest in the experimental facilities that allow the research community to address new threats, exploit emerging technologies, enable new applications, and foster the embedding of the network throughout the physical world. The alternative is unthinkable: an Internet that mediates most communication in our society, yet whose scale and importance makes it unalterable by innovative research, piloted by short-term commercial pressures, unmanageable even with the best intentions of huge numbers of network engineers, and still vulnerable to widespread outages and systematic attack.

The facility we propose is ambitious. Unlike traditional network testbeds that demonstrate a single design point, GENI is a global, general-purpose facility that places essentially no limits on the network architectures, services, and applications that can be evaluated. Unlike traditional network testbeds that either limit researchers to incremental changes or limit researchers to synthetic workloads, GENI is designed to allow both clean-slate designs and experimentation with real users under real-world conditions. Unlike traditional testbeds that provide no credible deployment path to the commercial world, GENI represents a model in which incremental adoption of new services drives wide-spread deployment.

Virtualization and programmabililty of the underlying substrate are the key enablers: they allow multiple network architectures and services to run simultaneously; they allow clean-slate designs to run side-by-side with incremental experiments; and they allow long-running services to attract real users, which results in realistic evaluations and drives adoption and deployment. GENI's modular design and well-defined interfaces are also important: they accommodate federation, which allows other countries and research communities to "plug into" GENI, and they simplify the task of incorporating new building block technologies into GENI over time.

As an evolving facility, GENI's management plan is also critical. Our approach is based on the tried-and-true model used throughout the computer science research community. Although we start with a proven framework, much of the work defining the specific interfaces takes place in working groups, which consist of the researchers using GENI and the developers building GENI. The working groups are established by a Technical Advisory Board, which also sets policy, arbitrates design decisions, and establishes implementation priorities. These priorities are then put into action by the Program Management Office, which oversees contracts to the participating development teams. This process is continuous, with early usage by the research community driving the evolution of GENI's software modules and interfaces throughout the construction phase.

In summary, GENI will be a unique facility that provides an opportunity for the same research community that created the Internet in the first place, to now define a *Future Internet* that is able to meet the demands of the 21st Century.

References

[ACK04] B. Ackland, M. Bushnell, C. Rose, I. Seskar, D. Raychaudhuri, M. Tentzeris, J. Papapolymerou, J. Laskar, and T. Sizer, "NeTS-ProWiN: High performance cognitive radio platform with integrated physical and network layer capabilities", *granted by National Science Foundation*, September 2004.

[ALA99] C. Alaettinoglu, et. al., "Routing Policy Specification Language (RPSL)," RFC 2622, June 1999.

[AKA] Akamai, www.akamai.com

[AND01] D. Andersen, H. Balakrishnan, F. Kaashoek, R. Morris, "Resilient Overlay Networks," *Proceedings of ACM SOSP*, 2001.

[AND03] T. Anderson, T. Roscoe, and D. Weatherall, "Preventing Internet Denial-of-service with Capabilities," *Proceedings of HotNets-II*, 2003.

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

[ARU92] R. Arun, P. Venkataram, "Knowledge Based Trouble Shooting in Communication Networks," *Proceedings of Symposium on Intelligent Systems*, November 1992.

[BAL02] M. Balazinska, H. Balakrishnan, D. Karger, "INS/Twine: A Scalable Peer-to-Peer Architecture for Intentional Resource Discovery," *Proceedings of the 1st International Conference on Pervasive Computing*, 2002.

[BAL04] H. Balakrishnan, K. Lakshminarayanan, S. Ratnasamy, S. Shenker, I. Stoica, and M. Walfish, "A Layered Naming Architecture for the Internet," *Proceedings of ACM SIGCOMM*, 2004.

[BAN03] N. Banerjee, Wei Wu, S. K. Das, "Mobility support in wireless Internet", *IEEE Wireless Communications*, 10(5), October 2003

[BAR03] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, A. Warfield, "Xen and the Art of Virtualization," *Proceedings of ACM SOSP*, 2003.

[BAV04] A. Bavier, M. Bowman, B. Chun, D. Culler, S. Karlin, L. Peterson, T. Roscoe, T. Spalink, and M. Wawrzoniak. "Operating System Support for Planetary-Scale Network Services," *Proceedings of the 1st Symposium on Network System Design and Implementation (NSDI '04)* San Francisco, CA (March 2004).

[BER94] T. Berners-Lee, "Universal Resource Identifiers in WWW," RFC 1630, June 1994.

[BER00] Y. Bernet, et. al., "A Framework for Integrated Services Operation over DiffServ Networks," RFC 2998, November 2000.

[BER02] S. Berson, S. Dawson and R. Braden. "Evolution of an Active Networks Testbed," *Proceedings of the DARPA Active Networks Conference and Exposition*, 5/02.

[BER03] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing", *IEEE Pervasive Computing*, 2(1), 2003.

[BLU05] D. Blumenthal, J. Bowers, and C. Partridge, Editors. "The Future of Optical Communications: Understanding the Choices," NSF Workshop Report, April 2005.

[BOH03] M. Bohge and W. Trappe, "An authentication framework for hierarchical ad-hoc sensor networks", *Proceedings of the ACM Workshop on Wireless Security*, 2003.

[BOR01] N. Borisov, I. Goldberg, and D. Wagner. "Intercepting Mobile Communications: The Insecurity of 802.11", *Proceedings of MOBICOM 2001*, 2001.

[BRA94] B. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: An Overview," RFC 1633, June 1994.

[BRA97] B. Braden, et. al., "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification," RFC 2205, Sepetember 1997.

[BRA02] B. Braden, T. Faber, and M. Handley. "From Protocol Stack to Protocol Heap – Role-based Architecture", *First Workshop on Hot Topics in Networks (HOTNETS-I)*, 2002.

[BYE98] J. Byers, M. Luby, M. Mitzenmacher, A. Rege, "A Digital Fountain Approach to Reliable Distribution of Bulk Data," *Proceedings of ACM Sigcomm*, 1998.

[CAE05] M. Caesar, D. Caldwell, N. Feamster, J. Rexford, A. Shaikh, and J. van der Merwe, "Design and Implementation of a Routing Control Platform," *Proceedings of NSDI*, May 2005.

[CAO00] Z. Cao, Z. Wang, E. Zergura, "Performance of Hashing-Based Schemes for Internet Load Balancing," *Proceedings of IEEE Infocom*, 2000.

[CHU00] Y. Chu, S. Rao, and H. Zhang, "A Case for End System Multicast," *Proceedings of ACM Sigmetrics*, June 2000.

[CHU02] B. Chun, J. Lee, H. Weatherspoon, "Netbait: a Distributed Worm Detection Service," Project website, http://netbait.planet-lab.org, 2002.

[CLA89] D. Clark, "Policy Routing in Internet Protocols," RFC 1102, May 1989.

[CLA90] D. Clark and D. Tennenhouse, "Architectural Considerations for a New Generation of Protocols," *Proceedings of ACM Sigcomm*, 1990.

[CLA03] D. Clark, C. Partridge, J. Ramming, J. Wroclawski, "A Knowledge Plane for the Internet," *Proceedings of ACM Sigcomm*, 2003.

[CLA05] D. Clark, et. al. Making the world (of communication) a different place. January 2005. http://www.ir.bbn.com/~craig/e2e-vision.pdf

[COU03] D. S. J. De Couto, D. Aguayo, J. Bicket and R. Morris, "A high-throughput path metric for multi-hop wireless routing", *Proceedings Mobicom*, September 2003.

[CRO03] M. Crovella, E. Kolaczyk, "Graph Wavelets for Spatial Traffic Analysis," *Proceedings of IEEE Infocom*, 2003.

[CSTB99] Computer Science and Telecommunications Board (CSTB), National Research Council. *Funding a Revolution: Government Support for Computing Research.* National Academy Press, 1999.

[CUL04] D. Culler, D. Estrin, and M. Srivastava. "Overview of sensor networks". *IEEE Computer, Special Issue in Sensor Networks*, August 2004

[DAB01] F. Dabek, M. Kaashoek, D. Karger, R. Morris, I. Stoica, "Wide-area cooperative storage with CFS," *Proceedings of ACM SOSP*, 2001.

[DEC00] D. Decasper, Z. Dittia, G. Parulkar, and B. Platter, "Router Plugins: A Software Architecture for Next Generation Routers," *IEEE/ACM Transactions on Networking*, February 2000.

[DEE89] S. Deering, "Host Extensions for IP Multicasting", RFC 1112, August 1989.

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

[DIG] Digital Fountain, www.digitalfountain.com

[DOR02] A. Doria, F. Hellstrand, K. Sundell, T. Worster, "General Switch Management Protocol (GSMP) V3," RFC 3292, June 2002.

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

[EST99] D. Estrin, R. Govindan, J. Heidemann and S. Kumar, "Next Century Challenges: Scalable Coordination in Sensor Networks," *In Proceedings of the Fifth Annual International Conference on Mobile Computing and Networks (MobiCOM '99)*, August 1999, Seattle, Washington.

[FAR05] A. Farrel, J. Vasseur, J. Ash, "Path Computation Element (PCE) Architecture," Internet-Draft, March 2005.

[FER98] P. Ferguson, D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," RFC 2267.

[FEL00a] A. Feldmann, A. Greenburg, C. Lund, N. Reingold, J. Rexford, F. True, "NetScope: Traffic Engineering for IP Networks," *IEEE Network*, March / April 2000.

[FEL00b] A. Feldmann, A. Greenburg, C. Lund, N. Reingold, J. Rexford, "Deriving Traffic Demands for Operational IP Networks: Methodology and Experience," *Proceedings of ACM Sigcomm*, 2000.

[FEA02a] N. Feamster, J. Borkenhagen, J. Rexford, "Controlling the Impact of BGP Policy Changes on IP Traffic," *Proceedings of NANOG25*, 2002.

[FEA02b] N. Feamster, J. Rexford, "Network-wide BGP Route Prediction for Traffic Engineering," *Proceedings of ITCOM*, 2002.

[FLO01] S. Floyd and V. Paxson, "Difficulties in Simulating the Internet," *IEEE/ACM Transactions on Networking*, August 2001.

[FLO02] S. Floyd and E. Kohler, "Internet Research Needs Better Models," *Proceedings of the First Workshop on Hot Topics in Networks (HotNets-I)*, October 2002.

[FRA01] P. Francis and R. Gummadi, "IPNL: A NAT-Extended Internet Architecture," *Proceedings of ACM SIGCOMM*, 2001.

[FRE04] M. Freedman, E. Freudenthal, and D. Mazieres, "Democratizing Content Publication with Coral," *Proceedings of NSDI*, March 2004.

[GAN04] S. Ganu, L. Raju, B. Anepu, S. Zhao, I.Seskar and D. Raychaudhuri. "Architecture and prototyping of an 802.11-based self-organizing hierarchical ad-hoc wireless network (SOHAN)", *Proceedings of Fifteenth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2004.

[GRI01] M. Gritter and D. Cheriton, "An Architecture for Content Routing Support in the Internet", *Proceedings of USITS*, March 2001.

[GUS97] E. Gustafsson and G. Karlsson. "A Literature Survey on Traffic Dispersion," *IEEE Network 11(2)*, March / April 1997.

[HAN02] M. Handley, O. Hodson, E. Kohler, "XORP: An Open Platform for Network Research," *Proceedings of ACM HotNets-I*, October 2002.

[HEI99] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," *In Proceedings of the fifth annual ACM/IEEE international conference on Mobile computing and networking*, August 1999.

[HEI01] John Heidemann, Fabio Silva , Chalermek Intanagonwiwat, Ramesh Govindan , Deborah Estrin , Deepak Ganesan, "Building efficient wireless sensor networks with low-level naming", *Proceedings of the eighteenth ACM symposium on Operating systems principles*, October 2001.

[HEL03] S. Helal, B. Winkler, C. Lee, Y. Kaddourah, L. Ran, C. Giraldo and W. Mann, "Enabling location-aware pervasive computing applications for the elderly", *Proceedings of 1st IEEE Conference on Pervasive Computing and Communications (Per-Com)*, March 2003

[HIN03] R. Hinden, S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture," RFC 3513, April 2003.

[HJA00] G. Hjalmtysson, "The Pronto Platform – A Flexible Toolkit for Programming Networks using a Commodity Operating System," *Proceedings of OpenArch*, 2000.

[HOC93] J. Hochberg, et. al., "Nadir: An automated system for detecting network intrusion and misuse," *Computers & Security*, 12(3), 1993.

[HSI03] H. Hsieh, et. al., "A Receiver-Centric Transport Protocol for Mobile Hosts with Heterogeneous Wireless Interfaces," *Proceedings of Mobicom*, 2003.

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

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

[IGE00] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *Proceedings of MobiCOM*, August 2000.

[INT00] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *In Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM 2000),* August 2000, Boston, Massachusetts.

[INT02] C. Intanagonwiwa., D. Estrin., R. Govindan, AND HEIDEMANN, J., "Impact of network density on data aggregation in wireless sensor networks", *Proceedings of the 22nd International Conference on Distributed Computing Systems, Vienna, Austria. IEEE*, 2002

[IOA02] J. Ionnidis and S. Bellovin, "Implementing Pushback: Router-based Defense Against DDoS Attacks," *Proceedings of NDSS*, 2002.

[IVT] Intel Virtualization Technology project page. http://www.intel.com/technology/computing/vptech/

[JOH04] D. Johnson, C. Perkins and J. Arkko, "Mobility support in IPv6", RFC 3375, June 2004

[KAH95] R. Kahn and R. Wilensky, "A Framework for Distributed Digital Object Services," 1995.

[KAR00] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks", *In Proc. ACM/IEEE MobiCom*, August 2000, pp 243-254.

[KAR02] S. Karlin and L. Peterson, "VERA: An Extensible Router Architecture," *Computer Networks*, 38(3), 2002.

[KUR02] J. Kurose (editor). Report of the NSF Worksop on Network Research Testbeds. www.gaia.cs.umass.edu/testbed_workshop, 11/02.

[KAN04] J. Kannan, A. Kubota, K. Lakshminarayanan, I. Stoica and K. Wehrle, "Supporting Legacy Applications over *i3*," UCB Technical Report No. UCB/CSD-04-1342, May 2004.

[KOH00] E. Kohler, "The Click Modular Router," Ph.D. thesis, MIT, November 2000.

[KON02] J. Kong, H. Luo, K. Xu, D. Gu, M. Gerla, and S. Lu, "Adaptive security for multi-layer ad-hoc networks," *Special Issue of Wireless Communications and Mobile Computing*, 2002.

[KUB00] J. Kubiatowicz, et. al., “Oceanstore: An Architecture for Global-Scale Persistent Storage,” *Proceedings of ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2000.

[KUB01] U. Kubach, and K. Rothermel. "Exploiting location information for infostation-based hoarding", *Proceedings of ACM Mobicom*, 2001

[GRU03] M. Gruteser, G. Schelle, A. Jain, R. Han, and D. Grunwald, "Privacy-aware location sensor networks", *Proceedings of Workshop on Hot Topics in Operating Systems (HotOS)*, 2003

[LAN03] Daniel Lang. A comprehensive overview about selected ad hoc networking routing protocols. *Technical report, Technische University München, Department of Computer Science*, March 2003.

[LAK04] A. Lakhina, M. Crovella, C. Diot, “Diagnosing Network-Wide Traffic Anomalies,” *Proceedings of ACM Sigcomm*, 2004.

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

[LEV02] P. Levis and D. Culler, "Mate: A Tiny Virtual Machine for Sensor Networks", *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*, Oct. 2002

[LI00] J. Li, J. Jannotti, D. De Couto, D. Karger, and R. Morris, "A scalable location service for geographic ad-hoc routing", *Proceedings of ACM MobiCom*, August 2000.

[LIS05] B. Liskov, A. Joseph, and F. Kaashoek, Editors. "Grand Challenges in Distributed Systems," NSF Workshop Report, September 2005 (pending).

[MIN05] G. Minden, "KU Agile Radio Technology", http://www.ittc.ku.edu/techreview2005/presentations/Minden_Agile%20Radios.ppt

[MIT99] J. Mitola III and G.Q. Maguire, "Cognitive radio: making software radios more personal", *IEEE Personal Communications*, August 1999.

[MOS05] R. Moskowitz, P. Nikander, P. Jokela, T. Henderson, “Host Identity Protocol,” Internet Draft, Feb. 2005.

[NEON] NEON Project. Report on the Second Meeting of the NEON Consortium Design Committee (CDC).

[NG01] E. Ng, I. Stoica, and H. Zhang, “A Waypoint Service Approach to Connect Heterogeneous Internet Address Spaces,” *USENIX Annual Technical Conference*, 2001.

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

[PAR93] C. Partridge, T. Mendez, and W. Milliken. “Host Anycasting Service,” RFC1546, November 1993.

[PAR04] K. Park, V. Pai, L. Peterson, Z. Wang, “CoDNS: Improving DNS Performance and Reliability via Cooperative Lookups,” *Proceedings of Symposium on Operating Systems Design and Implementation (OSDI)*, 2004.

[PAU03] G. Pau, D. Maniezzo, S. Das, Y. Lim, J. Pyon, H. Yu, M. Gerla, "A cross-layer framework for wireless LAN QoS support", *Proceedings of IEEE ITRE*, August, 2003.

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

[PER05] A. Perrig, D. Clark, and S. Bellovin, Editors. "Secure Next Generation Internet," NSF Workshop Report, July 2005.

[PERR02] A. Perrig, R. Szewczyk, D. Tygar, V. Wen, and D. Culler, "SPINS: Security protocols for sensor networks", *Wireless Networks*, (8)5, 2002.

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

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

[POO00] R. Poor. “Gradient Routing in Ad-hoc Networks”, MIT Media Laboratory, 2000.

[POS81] J. Postel, ed., “Internet Protocol,” RFC 791, September 1981.

[POR97] P. Porras, P. Neumann, “EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances,” *Proceedings of NIST-NCSC National Information Systems Security Conference*, 1997.

[PRE05] President’s Information Technology Advisory Committee. Cyber Security: A Crisis of Prioritization, February 2005.

[PRZ03] B. Przydatek, D. Song, A. Perrig, "Compression & aggregation: SIA: secure information aggregation in sensor networks", *Proceedings of the 1st international conference on Embedded networked sensor systems*, November 2003

[QUI01] B. Quinn, K. Almeroth, “IP Multicast Applications: Challenges and Solutions,” RFC 3170, September 2001.

[RAT02] S. Ratnasamy, M. Handley, R. Karp, S. Shenker, “Topologically-Aware Overlay Construction and Server Selection,” *Proceedings of IEEE Infocom*, 2002.

[RAT02b] Sylvia Ratnasamy, Deborah Estrin, Ramesh Govindan, Brad Karp, Scott Shenker, Li Yin, and Fang Yu, "Data-centric storage in Sensor-nets", *Proceedings of the First Workshop on Sensor Networks and Applications (WSNA)*, September 2002.

[RAY05a] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh, “Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols,” *IEEE WCNC*, 2005.

[RAY05b] D. Raychaudhuri and M. Gerla, Editors. “Wireless/Mobile Planning Group,” *NSF Workshop Report*, August 2005.

[RED99] R. Reddy, D. Estrin, R. Govindan, “Large-Scale Fault Isolation,” *IEEE Journal on Selected Areas in Communications*, March 1999.

[REX04] J. Rexford, A. Greenberg, G. Hjalmtysson, D. Maltz, A. Myers, G. Xie, J. Zhan, and H. Zhang, “Network-wide Decision Making: Toward a Wafer-thin Control Plane,” *Proceedings of HotNets-III*, November 2004.

[ROW01a] A. Rowstron, P. Druschel, “Storage Management and Caching in PAST, A Large-Scale Persistent Peer-to-Peer Storage Utility,” *Proceedings of ACM SOSP*, 2001.

[ROW01b] A. Rowstron, P. Druschel, “Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems,” *Proceedings of IFIP/ACM International Conference on Distributed Systems Platforms*, 2001.

[SAV99] S. Savage, A. Collins, E. Hoffman, J. Snell, T. Anderson, “The End-to-End Effects of Internet Path Selection,” *Proceedings of ACM Sigcomm*, 1999.

[SHE97] S. Shenker, C. Partridge, R. Guerin, “Specification of Guaranteed Quality of Service,” RFC 2212, September 1997.

[SHA99] A. Shaikh, J. Rexford, K. Shin, “Load-Sensitive Routing of Long-Lived IP Flows,” *Proceedings of ACM Sigcomm*, 1999.

[SNO02] A. Snoeren, C. Partridge, L. Sanchez, C. Jones, F. Tchakountio, B. Schwartz, S. Kent, and W. Strayer, “Single-Packet IP Traceback,” *IEEE/ACM Transactions on Networking*, 10(6), 2002.

[SNO04] A. Snoeren and B. Raghavan, “Decoupling Policy from Mechanism in Internet Routing,” *ACM SIGCOMM Computer Communication Review*, 34(1), 2004.

[SOU04] Eduardo Souto, Germano Guimaraes, Glauco Vasconcelos, Mardoqueu Vieira, Nelson Rosa, Carlos Ferraz, "A message-oriented middleware for sensor networks," *Proceedings of the 2nd workshop on Middleware for pervasive and ad-hoc computing*, October 2004

[SPR02] N. Spring, R. Mahajan, and D. Wetherall, “Measuring ISP Topologies with RocketFuel,”*Proceedings of ACM Sigcomm*, August 2002.

[SPR03] N. Spring, D. Wetherall, T. Anderson, “Scriptroute: A facility for distrubted Internet measurement,” *Proceedings of USENIX Symposium on Internet Technologies*, 2003.

[SRI00] C. Srisathapornphat, C.Jaikaeo and C.Chien-Chung Shen, "Sensor Information Networking Architecture", *International Workshops on Parallel Processing*, Pages 23-30, 2000.

[STE93] R. Steenstrup, “Inter-Domain Policy Routing Protocol Specification: Version 1,” RFC 1479, July 1993.

[STO01] I. Stoica, R. Morris, D. Karger, F. Kaashoek, H. Balakrishnan, “Chord: A Peer-to-Peer Lookup Service for Internet Applications,” *Proceedings of ACM Sigcomm*, 2001.

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

[SUB02] L. Subramanian, I. Stoica, H. Balakrishnan, R. Katz, “OverQoS: Offering Internet QoS Using Overlays,” *Proceedings of ACM HotNets*, 2002.

[SZE00] J. Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, and Kristofer Pister, "System architecture directions for networked sensors," in *Proceedings of the ninth international conference on Architectural support for programming languages and operating systems.* 2000, pp. 93--104, ACM Press

[TOU01] J. Touch. “Dynamic Internet Overlay Deployment and Management Using the X-Bone,” *Computer Networks*, July 2001, pp. 117-135.

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

[TOU03] J. Touch, Y. Wang, L. Eggert, G. Finn. “Virtual Internet Architecture,” ISI Technical Report ISI-TR-2003-570, March 2003.

[TOUM03] S. Toumpis and A. J. Goldsmith, "Performance, Optimization, and Cross-Layer Design of media access protocols for wireless ad-hoc networks", *Proceedings of the IEEE ICC 2003*, 2003.

[TUL01] P. Tullmann, M. Hibler, J. Lepreau, "Janos: A Java-oriented OS for Active Networks," *IEEE Journal on Selected Areas in Communications*. Volume 19, Number 3, March 2001.

[WAIL] Wisconsin Advanced Internet Laboratory project site. http://www.schooner.wail.wisc.edu/

[WAN02] L. Wang, V. Pai, and L. Peterson, "The Effectiveness of Request Redirection on CDN Robustness," *Proceedings of OSDI,* December 2002.

[WHI02] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb and A. Joglekar. "An Integrated Experimental Environment for Distributed Systems and Networks," *Proceedings of OSDI*, 12/02.

[WHY] WHYNET project page. http://chenyen.cs.ucla.edu/projects/whynet/

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

[YAN03] Xiaowei Yang, "Nira: A New Internet Routing Architecture,"*ACM SIGCOMM Computer Communication Review,* 33(4), 2003.

[YAN04] L. Yang, R. Dantu, T. Anderson, R. Gopal, "Forwarding and Control Element Separation (ForCES) Framework," RFC 3746, April 2004.

[ZHA04] M. Zhang, C. Zhang, V. Pai, L. Peterson, R. Wang, "PlanetSeer: Internet Path Failure Monitoring and Characterization in Wide-Area Services," *Proceedings of Symposium on Operating Systems Design and Implementation (OSDI),* 2004.

Appendix A: Work Chart

Appendix B: Budget