# G E N I

Global Environment for Network Innovations

# Lifecycle of a GENI Experiment

Document ID: GENI-SE-SY-TS-UC-LC-01.1

25 February 2009

TABLE OF CONTENTS

## 1  Document Scope

This section describes this document's purpose, its context within the overall GENI document tree, the set of related documents, and this document's revision history.

### 1.1  Purpose of this Document

This document describes an experimenter's view of the lifecycle of an experiment in the GENI environment.  The purpose of this document is to understand the workflow of the experimenter and identify tools and services needed to simplify experimentation in GENI.

### 1.2  Context for this Document

Figure 1 below shows the context for this document within GENI's overall document tree.
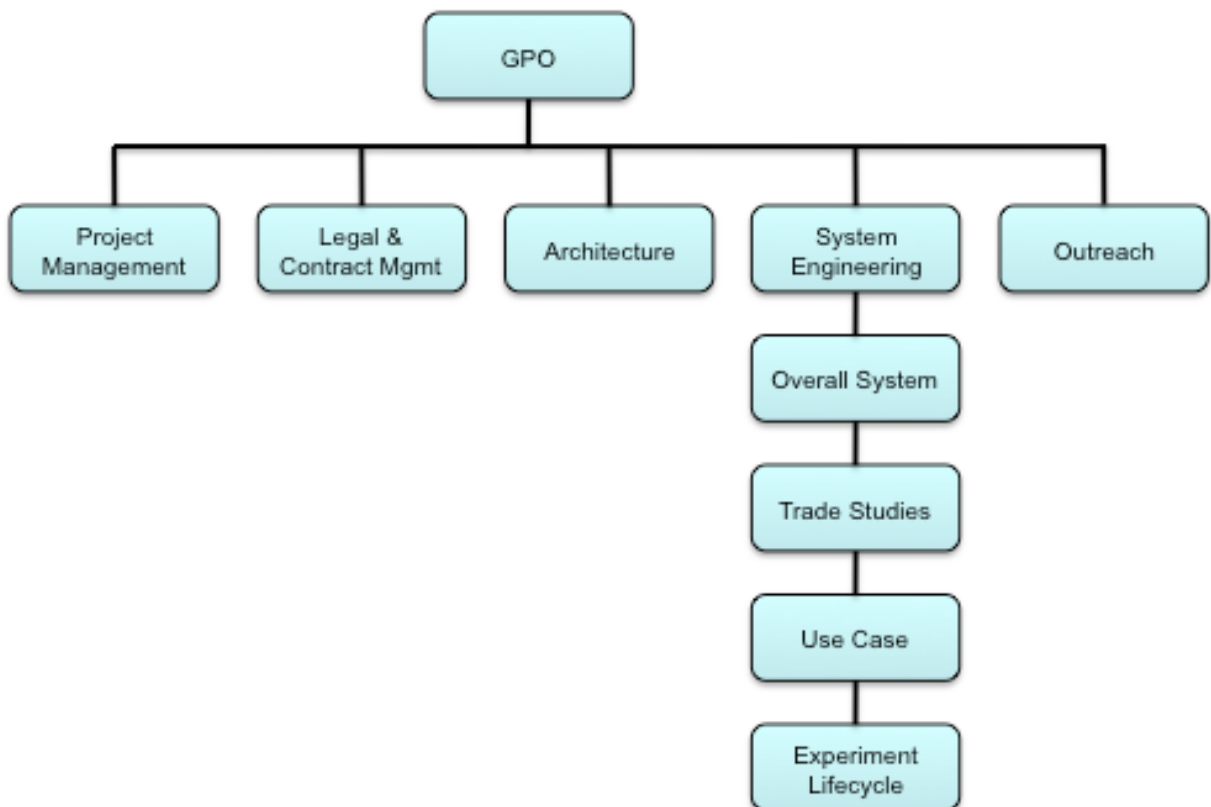


Figure 1. This document within the GENI Document Tree.

### 1.3  Related Documents

The following documents of exact date listed are related to this document, and provide background information, requirements, etc., that are important for this document.

### 1.3.1   National Science Foundation (NSF) Documents

| Document ID | Document Title and Issue Date |
|---|---|
|  |  |

### 1.3.2   GENI Documents

| Document ID | Document Title and Issue Date |
|---|---|
| GENI-SE-SY-SO-02.0 | GENI System Overview (September 29, 2008) |

### 1.3.3   Standards Documents

| Document ID | Document Title and Issue Date |
|---|---|
| N / A |  |

### 1.3.4   Other Documents

| Document ID | Document Title and Issue Date |
|---|---|
| N / A |  |

### 1.4   Document Revision History

The following table provides the revision history for this document, summarizing the date at which it was revised, who revised it, and a brief summary of the changes. This list is maintained in reverse chronological order so the newest revision comes first in the list.

| Revision | Date | Revised By | Summary of Changes |
|---|---|---|---|
| V. 1.0 | 19 Jan 2009 | V. Thomas | Initial draft. |
| V. 1.1 | 25 Feb 2009 | V. Thomas | Added a usage narrative section. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 2   GENI Overview

The Global Environment for Network Innovations (GENI) is an ambitious effort to create a large-scale experimental network research facility that will allow researchers to escape today's Internet-circumscribed research environment.

GENI is envisioned as a national-scale research facility for network science and engineering, which will then grow and federate worldwide to support millions or perhaps billions of devices. It will allow controlled end-to-end experimentation, at large scale, of new global computing / communications architectures, which may or may not be compatible with the Internet.

Unlike traditional network testbeds, GENI is conceived as a general-purpose facility that should place 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 widespread deployment.

It will support a wide range of experimental protocols, and data dissemination techniques; running over a substantial fiber optic infrastructure with next-generation optical switches, novel high-speed routers, city-wide experimental urban radio networks, high-end computational clusters, and sensor grids. All infrastructures can be shared among a large number of individual, simultaneous experiments with extensive instrumentation that makes it easy to collect, analyze, and share real measurements.

Core concepts for GENI are as follows:

- **Programmability** – researchers may download software into GENI nodes to control how those nodes behave;

- **Virtualization and Other Forms of Resource Sharing** – whenever feasible, nodes implement virtual machines, which allow multiple researchers to simultaneously share the infrastructure; and each experiment runs within its own, isolated slice created end-to-end across the experiment's GENI resources;

- **Federation** – Different parts of GENI are owned and/or operated by different organizations, and the NSF portion of GENI forms only a part of the overall 'ecosystem'; and

- **Slice-based Experimentation** – GENI experiments will be an interconnected set of reserved resources on platforms in diverse locations.  Researchers will remotely discover, reserve, configure, program, debug, operate, manage, and teardown distributed systems established on the GENI substrate.

## 3    Document Overview

This document describes an experimenter's view of the lifecycle of an experiment in the GENI environment.  The objective of this document is to understand the workflow of the experimenter and identify tools and services needed to simplify experimentation in GENI.

An experiment is a researcher-defined use of a slice; an experiment runs in a slice.  A slice is a substrate-wide network of computing and communication resources capable of running an experiment or a wide-area network service. Many different experiments can run in a particular slice concurrently or over time.

It is hoped GENI experimenter tools and services will hide details of slices and other GENI control framework constructs from most researchers.  The remainder of this document intentionally does not use terms such as slices or slivers.  Rather, terms such as experiments and resources that are more natural to the experimenter are used.  Resources include computation, communication, measurement, and storage resources.  They can be contained in a single physical device or distributed across a set of devices.

GENI will have users with a wide range of abilities and needs.  At one end of the spectrum are users such as undergraduates using GENI for class projects.  These users need tools and services that make it easy to quickly set up and run short-lived experiments.  At the other end of the spectrum are researchers who need a great deal of control over their experiments or over the data collected by the experiments.  These users may need large numbers of very specific types of resources and may need them for extended periods of time.  Such experiments will need to tolerant to changes in resources available to them as these resources may fail, be taken down for maintenance or upgraded.

The workflow described in this document is largely independent of the type of GENI user.  The tools and services developed to support this workflow must however be cognizant of the different needs of these users.

The document introduces the experiment lifecycle using a *usage narrative*, a fictional story that describes an experiment as it plays out on GENI (Section 4).  The purpose of this narrative is to help identify the phases in the lifecycle of an experiment and the tools needed to support them.

Sections 5 through 5.5 describe the experiment lifecycle phases identified to date.  These lifecycle phases are shown as green boxes in Figure 2.  The figure shows a high-level view of the lifecycle of a GENI experiment.  The boxes in the figure represent different phases of the lifecycle.  Even though these phases are laid out sequentially for convenience of presentation it is understood that some experiments may cycle between phases or may skip some phases.

The first two phases of the lifecycle shown in Figure 2 are *Experiment Design* and *Software and Hardware Development*.  Experiment design is related to formulating a hypothesis to be tested, identifying controlled and uncontrolled variables and developing a system for measuring system responses that validate or invalidate the hypothesis.  The next phase, software and hardware development, occurs in the lab and serves to validate the experiment design on a small scale.  This lab phase may be skipped if the experiment requires resources that are only available in GENI.  Both these phases are outside the scope of GENI and are therefore not elaborated on further in this document.

*Experimenter Registration* is related to authenticating users and giving them appropriate GENI credentials.  This lifecycle phase is described in greater detail in Section 5.
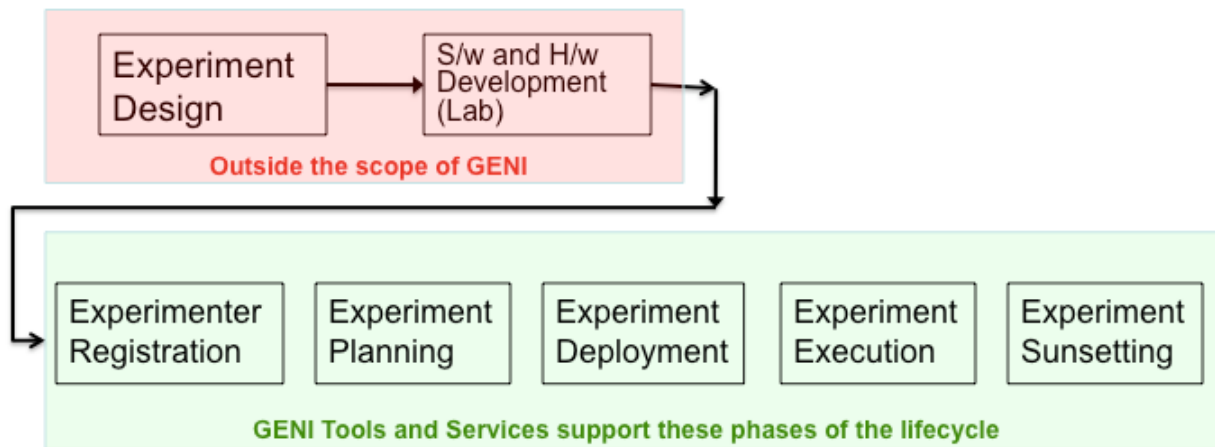
Figure 2. Top-level view of the lifecycle of a GENI experiment.


*Experiment Planning* is the phase where the user plans how the experiment will be conducted in the GENI environment: Determine the resources needed and tools needed to program these resources, the instrumentation services available and their setup, etc.. This phase is described in Section 5.2

*Experiment Deployment* is related to actually obtaining GENI resources and installing the software and hardware needed to run the experiment. This is described in Section 5.3.

*Experiment Execution* is the actual running of the experiment: Starting, pausing, resetting, growing, shrinking and stopping the experiment. This phase is described in Section 5.4.

*Experiment Sunsetting* is related to archiving the experiment so it is available to other experiments to repeat, modify or extend. It also includes transition of technology demonstrated by the experiment to product. This is described in Section 5.5.

Finally Section 6 describes other GENI services that support experimentation.


## 4    A GENI Usage Narrative

This GENI usage narrative is a story meant to illustrate the use of the GENI testbed. In some places the story assumes a certain design of the GENI control framework even though aspects of this design are currently being debated. This usage narrative will be modified in the future as the control framework design decisions are finalized.

Prof. Adams of Big State University has been working with his graduate student Priya on a protocol suite that can better support mobile cellular network users accessing the Internet from smartphones and other devices. The protocol suite includes new routing schemes centered around cellular base station locations, networking protocols that incorporate knowledge held by cellular signaling protocols, and fast proxy deployments based on predictions of user location by mobility models.

Simple analysis and small-scale simulation studies have shown Prof. Adams' protocol suite might provide mobile Internet users with consistently faster and better Internet access compared to the existing Internet protocol suite. This needs to be verified experimentally on a scale that can only be provided by the GENI networking research testbed.

## 4.1   Experiment Planning

Prof. Adams and Priya decide to evaluate their protocol suite by deploying it on a simulated cellular network in a metropolitan area.  They estimate such a network for a single provider would consist of about 20 cell base stations and 2,500 simultaneous users.

Prof. Adams and Priya determine the GENI testbed is well suited for simulating such a network: It is capable of supporting large scale experiments, is readily accessible, and is well regarded as a realistic testbed for scientific experimentation.

### 4.1.1   Resource and Tool Discovery

Both Prof. Adams and Priya are registered GENI experimenters.  They possess cryptographically signed credentials issued by their university that identify them as faculty and graduate student respectively.

Priya accesses the GENI testbed using a web browser on her desktop.  She signs on to the testbed using her GENI provide certificate.  Her certificate is verified.

Priya then uses GENI's *keyword based search engine* to search GENI's repository for experiments that might have simulated a cellular network on GENI.  She finds an experiment that used a network of four cellular base stations to evaluate different mobile device handoff protocols.  She downloads from the archive a paper describing the experiment, a formal GENI *Experiment Specification Language* description of the experiment and archived measurements.

Priya determines she can modify the downloaded experiment specification for her experiment.  She makes the necessary changes but retains the four-base station sized network in the original experiment to make it easier to debug her experiment.

The experiment specification includes a description of resources needed by the experiment.  In this case it specifies four nodes with significant compute capabilities to simulate the base stations, a wired network with two software routers to connect the base stations, and 100 nodes with relatively small compute capabilities to simulate mobile nodes.

Priya then uses a GENI *resource discovery tool* to find resources for her experiment.  The tool parses the experiment specification and finds candidate resources for the experiment.  It presents Priya with a list of these resources along with their location, the backbone network they connect to, their compute capabilities, and number and bandwidth of network interfaces.  Priya decides some of these resources allocated for the base station nodes would not be suitable for her experiment as they connected to the Internet via different backbone networks.  For realism she wanted all base station nodes to be on the same backbone network.

Priya then modifies the experiment specification to constrain all resources for the base stations to a single backbone network.  The resource discovery tool finds a new set of resources that meet Priya's needs.  She "bookmarks" these resources in the discovery tool so she can request them at a later time.  She then logs out of GENI.

### 4.1.2   Developing a GENI Version of the Experiment

Now that Priya is confident she can test their protocol suite on GENI, she implements this suite within the Linux operating system kernel.  She develops three operating system images for the three types of nodes on her network.

The operating system images developed for the base station and the mobile nodes include link quality estimation modules that estimate and adjust the link quality between a mobile node and all visible base stations.  This is needed as node mobility is being simulated in this experiment.  The link

quality modules use simple free-space RF propagation models configured with the appropriate transmitter powers and receiver sensitivities. Other software on the base station and mobile node that implement the standard cellular phone hand-off protocol base their decisions on estimates from the link quality estimation modules.

The software for the mobile nodes also includes a mobility model. Priya downloads and reuses the random mobility model software from the cellular network experiment she had discovered earlier in the GENI archive.

Finally, Priya adapts and includes two standard *GENI packages* to her software. The first is an experiment control client package that she adapts to the specific needs of her experiment. This client listens to commands from a GENI experiment control tool and takes actions that are appropriate for the experiment. For example, a command to pause the experiment sent to a mobile node would cause the node to stop its simulated motions and remain connected to its current base station.

The second standard GENI package she includes is an instrumentation package. This package takes real-time instrumentation data generated by an experiment, packages it into a standard GENI instrumentation format and ships it over the GENI instrumentation plane to viewers and log files as directed by the GENI experiment control tool.

## 4.2    Experiment Deployment

### 4.2.1   Leasing Resources

Priya is now ready to evaluate her protocol suite on GENI. She logs back on to GENI, creates a GENI experiment and gives it a name. Since she created the experiment she has all privileges over the experiment including the ability to add researchers to the experiment, control the privileges they have, and start and stop the experiment. Priya adds Prof. Adams as a researcher on the experiment and gives him full privileges over the experiment. To add Prof. Adams she uses the GENI *search engine* to search for principals with the last name Adams and who belong to Big State University. She selects her advisor for the list of principals returned by the search engine and adds him to the experiment.

Priya then uses the GENI *resource discovery tool* to obtains leases for the resources she had discovered and bookmarked earlier. She leases them for five days starting immediately. (Five days is the default maximum duration for which a graduate student can lease resources without her faculty advisor explicitly raising this limit.)

Her experiment specification includes a "label" for each of the three types of resources she has requested: base station node, router node and mobile node. The resources leased are automatically labeled appropriately.

### 4.2.2   Setting up the Network

The *resource discovery tool* informs Priya that all resources she requested are available and her leases have been granted; she can start using them immediately. She switches to a *network definition tool* to set up the network for her experiment. Since she has a relatively small number of base station and router nodes connected by a wired network, she chooses to set this up manually using a graphical tool. Her other option is to use a *network description language* to parametrically specify the network topology and other characteristics of the network.

Priya uses the *network definition tool's* graphical interface to set up wired connections between the nodes representing cellular base stations and the router nodes. She mistakenly asks for 1,000 Mbps links between these nodes. The tool detects the mistake since it knows from the specification of these

resources that the fastest link they can support is 100Mbps.  Priya corrects her mistake.  She then uses the tool to set up the "mobile nodes".

Priya plans to simulate node mobility and to emulate a wireless cellular network over the wired links.  To do this she specifies direct connectivity between every mobile node to every base station. Her plan includes the use RF propagation models to estimate signal strengths between base stations and mobile nodes if this had been a wireless network, use standard cellular network protocols to determine which base station each mobile node must connect to, and use the corresponding wired link to communicate between the mobile node and base station.  As the nodes "move" in this simulated environment, their estimated link quality to the different base stations will change, they will be handed-off to other base stations, and Priya's software will use a different link to talk to its new base station.

Figure 3 shows a snapshot of the configuration of the network.  The solid lines connecting the base stations and routers are wired connections in the tesbed and in the experiment.  The dashed lines between the mobile nodes and the base stations are wired connections in the testbed used to emulate wireless connections.  Note that every mobile node is using only one of its base station connections at any one time, shown in the figure as a dark dashed line.

### 4.2.3   Verifying the Network Setup

Priya decides to verify the network is setup correctly and everything works as expected using the IP protocol stack before she downloads the protocol stack she has been developing with Prof. Adams.  She uses a GENI *provisioning tool* to load a standard Linux distribution into the base station nodes and router nodes and a standard embedded Linux distribution into the mobile nodes.  The *provisioning tool* understands IP networks and so Priya uses it to automatically assign IP addresses to all network interfaces.
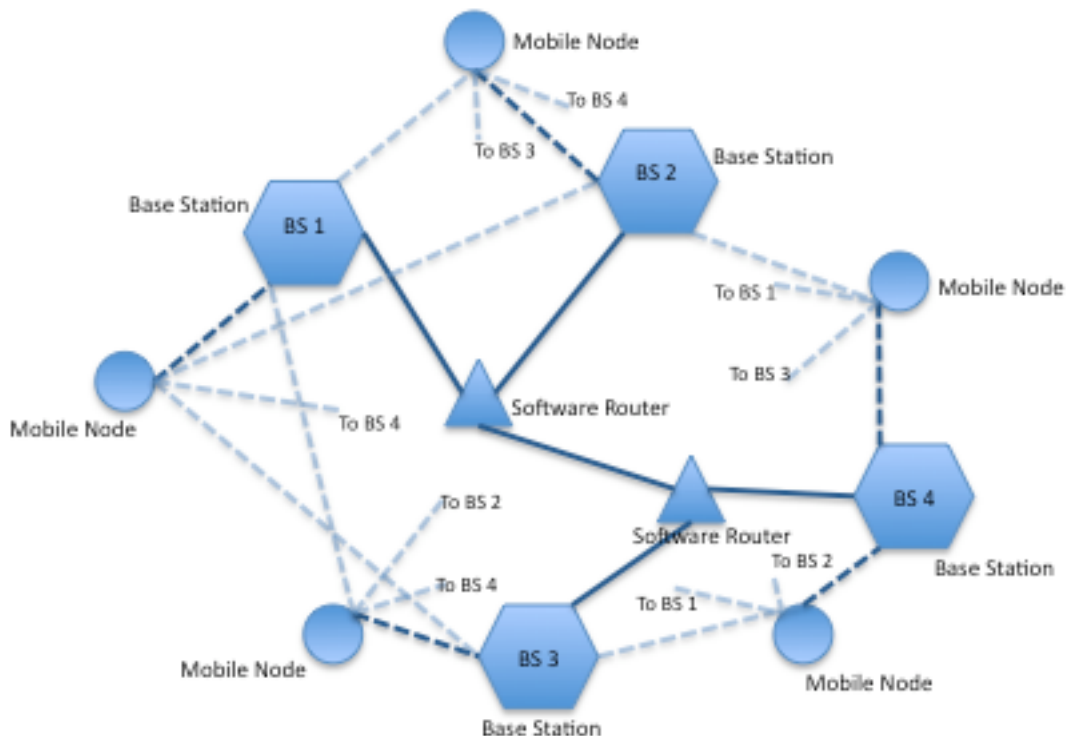


Figure 3. Network configuration for this experiment.

Priya then downloads a simple "ping" program from the *GENI Users Group (GUG)* software repository.  The GUG also maintains a web site that includes helpful hints and utilities provided by GENI users and a discussion group for users to share experiences, help one another and discuss features they'd like included in future versions of GENI.

The "ping" program is useful for verifying node connectivity.  Priya uses the GENI *provisioning tool* to download this ping program into the base station and mobile nodes.  She then uses this tool to load a configuration file into each of the nodes: this configuration file describes the topology of the network and is used to by the ping program to ensure the node to talk to every other node in the network.

Priya then uses the GENI *experiment control tool* to specify output from the ping program be directed to a log file on the local host.  This log file is created in the slice of each host's file system created for Priya.  She then uses this *experiment control tool* to start the execution of the ping program on all of the nodes.  She lets the experiment run for about five minutes and then stops execution of her program using the *experiment control tool*.

Priya then uses the *instrumentation control view* of the *experiment control tool* to retrieve the logged data and analyze it.  She finds the network did indeed get set up as expected i.e. the nodes are able to communicate and network throughput and latencies are as expected.

### 4.2.4   Deploying the Experiment

Priya is now ready to deploy her experiment.  She uses the same *provisioning tool* mentioned in the previous section to install the software she has developed.  Since the three types of nodes in her network were labeled during network setup, she can easily specify the software image that gets loaded into each type of node.

In addition to software, she uses the *provisioning tool* to install configuration files for use by the software such as the link quality estimation module.

### 4.3   Experiment Execution

### 4.3.1   Running the Experiment

The experiment is now ready to run.  Priya starts up the GENI *experiment control tool* that connects to her experiment.  The tool ensures it can connect to the experiment controller clients Priya included in the software loaded into the network nodes.  The tool's console displays a status of the experiment: The number of nodes in the experiment and the state of the experiment on the nodes (running, paused, stopped, unreachable, failed, etc).  It provides options for "drilling-down" into any one node to examine the status of the experiment there, the versions of software loaded, and machine specific information such as processor load, memory and network utilization and file system sizes.

Priya then uses the GENI *experiment control tool* to initialize her experiment in a phased manner. She first initializes the base station and router nodes.  Once they have discovered one another, the routing tables in the routers have been populated, and the base stations are able to communicate, she initializes the mobile nodes.  Each of these nodes picks a base station to connect to based on signal strength estimates from the link quality estimation module.

She then uses the GENI *experiment control tool* to start the experiment.  The start command causes the mobile nodes to move according to their mobility model and to generate Internet traffic based on a pre-programmed statistical model.  As the experiment runs it logs information of interest to the GENI instrumentation plane.  Some of the information to logged is explicitly provided by the software as it

executes.  Other information such as traffic flows is collected by GENI and logged because Priya instructed the GENI instrumentation service to do so.

### 4.3.2   Instrumentation and Measurement

Experiment data logged to the GENI instrumentation plane is cached on each node's local storage. Priya switches to the GENI *instrumentation control view* in her *experiment control tool*.  She uses the tool to selectively stream the instrumentation data from just the mobile nodes to her display.  She watches, in real-time, the traffic being generated by each of the nodes, the network bandwidths and latencies they observe.  She likes what she sees, sends an email to her advisor Prof. Adams about the status of the experiment, and heads home leaving the experiment running.

Later that night, Prof. Adams sees the email from Priya about the experiment.  He logs in to GENI and is presented a list of experiments that have him listed as a participating researcher.  He finds the experiment created by Priya and selects it.  GENI sees he has all privileges on the experiment, starts up an *experiment control tool* that connects to the experiment and presents him with *the instrumentation control view*.  Prof. Adams has the tool display instrumentation data from the mobile nodes.

He studies this data for a few minutes and notices occasional intervals when a number of consecutive packets to a mobile device are dropped.  He stops the streaming of  the data from the mobile devices and scrolls back to a time where he noticed this period of dropped packets.  He then uses the same *instrumentation control tool* to retrieve traffic logs from around that period from the base station and router nodes.  He examines these logs and discovers a router is dropping packets during those times.  He theorizes this may be due to a buffer overflows at the base station and sends an email to Priya asking her to investigate this further.

### 4.3.3   Debugging an Experiment

The next day Priya reads the email from Prof. Adams and starts to look into the issue of dropped packets.  She logs into GENI and uses the GENI *instrumentation control tool* to look for the periods of dropped packets that Prof. Adams observed.  She finds a few such periods and selects one of them to investigate.  She examines all traffic to and from the mobile node around this period, the base station it was connected to and the routers in the network.  She too notices packets being dropped by a router during this time.

She further investigates the cause of the dropped packets and determines there are periods when the router is unable to keep up with the processing of network traffic.  She decides to fix the problem temporarily by increasing buffer sizes on the router nodes.  She makes the necessary changes in her source code and recompiles it on her workstation.  She then pauses her experiment using the GENI *experiment control tool*, which causes all nodes to enter a quiescent state where they don't send messages or respond to messages.  She then shuts down the router nodes and uses the GENI *provisioning tool* to install the new version of her software.  She then switches to the GENI *experiment control tool* to start these router nodes and to un-pause the other nodes.   She goes back to her *instrumentation control view* to observe the experiment.  She does not see the dropped packet problem anymore but does notice periods of time with increased packet latencies.  This serves to validate her theory the routers are having trouble keeping up with packet processing during these time periods.

### 4.3.4   Changing Resource Allocations

Priya and Prof. Adams discuss this experiment and decide that the only way to reliably estimate the latencies and bandwidths seen by the mobile nodes is to increase the processing power of the routers.

They determine they need specialized routing hardware in order to be able to draw reliable conclusions from their experiment.   They decide to look into the use of FPGA-based routers.

Priya logs into GENI and uses the GENI *experiment control tool* to stop the experiment.  She then uses the tool's *instrumentation control view* to direct all data logged by her experiment to be copied over to her workstation.  She then terminates the experiment, thereby releasing the leases on the resources used by the experiment.

Priya then goes back to her experiment specification and edits it to require FPGA-based resources for the router nodes.  For the base station and mobile nodes she specifies by name the same resources she had leased earlier since her software had already been tested on those resources.  She then uses the GENI *resource discovery tool* to find resources that match her experiment specification.  The tool returns to her four FPGA-based routers as specified in her experiment specification and the specific resource she had requested for simulating base stations and mobile nodes.

Fortunately for Priya, all four FPGA-based routers are based on FPGAs from the same vendor.  The resource discovery tool also points her to the vendor's web site that includes tools and tips for programming these FPGAs.  Priya adds these FPGA-based routers to her list of "bookmarked" resources and logs out of GENI.

Priya spends the next three weeks developing and debugging an FPGA-based implementation of her protocol stack.  She uses a couple of development kits from the FPGA vendor for this purpose.  After three weeks of development and testing she is reasonably confident she can re-run her experiment on GENI with the FPGA-based routers.

Priya logs back into GENI, uses the GENI *resource discovery tool* to lease the resources she has bookmarked, the GENI *network definition tool* to set up the simulated network, the GENI *provisioning tool* to download appropriate software images and configurations into the nodes.  She then uses the GENI *experiment control tool* to initialize and run the experiment.  This time she does not see the periods of dropped or delayed packets she had seen earlier, indicating the FPGA-based routers were keeping up with packet processing.


### 4.3.5   Growing the Experiment

Prof. Adams and Priya are pleased with how well their protocol suite is performing in the simulated network.  They decided to increase the number of base station nodes and mobile nodes to 20 and 2,500 respectively to more accurately reflect a metropolitan cellular network.  They decide to keep the traffic generator for the mobile nodes they had used earlier i.e. all mobile nodes are assumed to constantly active and generating traffic according to a statistical model.

Priya logs back into GENI and connects to her experiment that is still running.  She goes back to her experiment specification and adds to it the new base stations, router nodes and mobile nodes.  The GENI *resource discovery tool* finds her the additional resources she needs; since she does this discovery in the context of a running experiment the tool is able to determine what resources she already has and what new ones are needed.  Likewise, the GENI *provisioning tool* can be told to install software into only those nodes that don't already have a certain version of the software already running.  This allows Priya to very quickly grow her experiment.

The experiment is a great success.  The results clearly show the new protocol suite greatly out-performs the existing Internet protocol suite.  Prof. Adams and Priya write a paper summarizing their research for presentation at a conference and a longer paper for publication in a journal.

### 4.3.6   Adding a Research Collaborator

Prof. Adams presents his new protocol suite and experimental results at a conference.  Attending this conference is Dr. Enoch, a researcher with the cellular service provider Peripatetic Technologies. Dr. Enoch is very interested in Prof. Adams' work and discusses with him the possibility of testing this protocol suite under even more realistic conditions.

Prof. Adams and Dr. Enoch decide GENI is still the best way to test this protocol suite.  They decide to start by deploying this protocol suite on a small number of experimental handsets used by company personnel.  They would start by adding two base stations to the experiment—these would be real base stations that direct this experimental traffic to the nodes in the experiment over Ethernet VLANs.  They also decide to add Internet gateways to the experiment so users can access real web sites.  This new experiment network configuration is shown in Figure 4.

Big State University and Peripatetic Technologies sign the necessary agreements to work together on this experiment and for Peripatetic Technologies to make available to the experiment a fraction of the compute and communication resources from two base stations.

Peripatetic Technologies also signs an agreement with GENI and registers with the GENI clearinghouse an aggregate consisting of two base stations.  As part of this agreement the aggregate is made visible only to researchers at Big State University i.e. other researchers cannot discover or lease resources from the aggregate.

Next, Prof. Adams registers Dr. Enoch with the GENI Principals registry.  Since Peripatetic Technologies is not registered with GENI as an identity provider, Dr. Enoch is registered as a researcher vouched for by Prof. Adams.  Dr. Enoch's access to GENI is limited to the experiment on which he is collaborating with Prof. Adams.  He can control the experiment because Prof. Adams gives him those privileges but he cannot add resources to the experiment because of GENI policy.

### 4.3.7   Adding Opt-In Users

Prof. Adams and Priya take their experiment specification and modify it for this new experiment. They add the two real base station nodes and annotate them as being external to GENI.  Since the real mobile nodes communicate only with the real base stations they are not part of the experiment specification.  They add two "protocol gateway" nodes to their experiment; these nodes act as protocol translators between Prof. Adams' protocol suite and the Internet protocol suite.  These translators are marked as being connected to the public Internet.

The researchers go back to GENI to create a new experiment.  This time they add Dr. Enoch as a collaborator on the experiment.  They then acquire the resources for their experiment—they request the same resources they had for the original experiment and the additional resources needed for this new experiment.  They use the GENI *network setup tool* to configure the VLANs on the GENI base station nodes.  Dr. Enoch manually configures the VLANs on his end.

Priya then uses the GENI *provisioning tool* to install appropriate versions of the software into the nodes in the experiment.  Dr. Enoch manually installs software on the real base station nodes.

With everything set up and network connectivity tested, Priya initializes and starts the experiment. Dr. Enoch initiates the first http request from his smartphone.  This request gets sent over Prof. Adams' protocol suite to one of the real base stations, on to the router connected to a protocol translator and finally on to the Internet.  Responses to this request take this path in reverse.

After a few days of testing, Dr. Enoch hands out a few smartphones with the new protocol suite to some of colleagues.  He tells them about the experiment and about the great improvement in his Internet browsing experience when he is connected to one of the two base stations.

The colleagues who try these smartphones are soon raving about their experience.  Word gets around the company and more people ask to try them out.  Dr. Enoch convinces his management to let him install this software on a few more base stations.  He also sets up his own protocol translator node that connects to the Internet so all Internet traffic does not have to go through GENI.

## 4.4   Transitioning to Product

As the popularity of the new Internet access service grows within Peripatetic Technologies, more and more of the company's base stations are provisioned with the new protocol suite.  Soon its marketing department decides to turn this into a premium service for its customers.  Peripatetic Technologies enters into a licensing agreement with Big State University that allows it to install this software into its infrastructure and allow paying customers to use for their web browsing.

Prof. Adams, Priya and Dr. Enoch publish a couple more papers on their experiences with this technology in the real-world and with transitioning this technology from a GENI experiment into a commercial product.



Figure 4. Experiment network configuration with real base stations and opt-in users.

Priya uses a GENI *archival tool* to archive the experiment. The tool leads her through the steps needed to complete the archival process: Brief information about the experiment, contact information for the experimenters, relevant keywords, funding source(s), experiment specification(s) used by the researchers, salient data sets, and papers/reports published about the technology.

## 5    Experiment Lifecycle: Phases and supporting Tools

This section describes the phases of a typical GENI experiment such as the one described above. Also described are tools and services required to support these phases. The above narrative not only provides the motivation of these tools and services but also serves to illustrate how they may be used.

### 5.1    Experiment Registration

Experimenters register with research organizations on the basis of existing or planned experiments. In essence, the research organization vouches that a particular experiment is indeed being planned or conducted, and that this particular individual is authorized to participate in the experiment as a researcher. Multiple researchers with different researchers organizations may collaborate on the same experiment. Experimenter registration enables trust relationships to be established among collaborators on an experiment: Research organizations vouch for their researchers, clearinghouses vouch for research organizations registered with them, and clearinghouses set up trust relationships amongst themselves. Experimenter registration also establishes relationships between researchers and experiments, and hence GENI slices.

Registered experimenters are given permissions to perform one or more tasks within the context of a GENI experiment. For example, some experimenters may be permitted to browse resources (GENI components) while others might be permitted to browse and lease resources for running experiments.

GENI clearinghouses may have policies related to experimenters registered with other clearinghouses. For example a GENI clearinghouse in Europe may permit users registered with the NSF clearinghouse in the US browse resources but not lease them.

### 5.1.1    Tools and Services for User Registration

GENI must provide administrative tools to register users and provide them the necessary credentials to use GENI.

### 5.2    Experiment Planning

Experiment planning in GENI includes the four steps shown in Figure 5. The *Present Credentials* phase is where the experimenter presents to GENI credentials issued during registration and is authenticated by GENI as being a registered user.
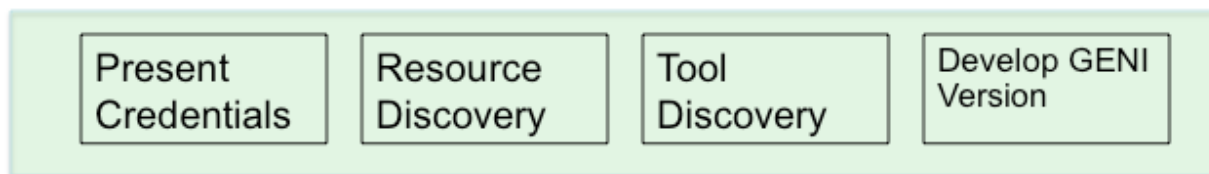


Figure 5. Steps in the Experiment Planning phase of the experiment lifecycle.

The experimenter then creates an experiment, which is an identifier for the context within which he or she will discover resources and control them. Experiments have names and unique identifiers.

Experiments may have multiple experimenters associated with them. Different experimenters may have different privileges with respect to that experiment such as the ability to add new experimenters and assign them privileges, the ability to add or remove resources to the experiment and the ability of view data collected by the experiment. Experimenters with different privileges may be added or removed at any time during the life of an experiment.

In the *Resource Discovery* phase the experimenter looks for the resources needed to run an experiment. Resource discovery can happen in one of many ways:

1. Automatically from an experiment specification. GENI will support one or more *experiment specification languages*. This language is used to describe attributes of an experiment including resources needed, configuration of these resources and data to be collected during the execution of the experiment.

2. Programmatically. The experimenter must be able to write a script or program that queries clearinghouses for resources.

3. From an interactive user interface. Experimenters must be able to browse the clearinghouses for resources using an interactive interface such as a shell-like environment or a graphical environment.

For scalability, resource discovery might be a multi-step process with the experimenter initially getting course information about resource aggregates and then being able to refine queries to get more detailed information on specific resources or aggregates.

GENI may also permit experimenters to lease resources as they browse them, if the experimenter is permitted to lease resources.

Once resources needed to run the experiment are discovered, the experimenter will need to learn about the tools needed to program and configure these resources. This happens in the *Tool Discovery* step. In this step experimenters may learn about constraints to be accounted for when programming and using these resources. Tool discovery may also point to emulators for specialized resources that can be used to develop and validate software before it is deployed on GENI.

Finally, in the *Develop GENI Version* step the experimenter develops the software and configuration files that will eventually be loaded into the resources to be used for the experiment. This software may be tested in a lab, using emulators for specialized resources or on GENI itself.

### 5.2.1   Tools and Services for Experiment Planning

GENI must provide a tool that can interpret an experiment specification and query appropriate clearinghouses for resources.

GENI clearinghouses must provide an API that can be used to query for resources. Software libraries that make it easy to invoke these libraries over a network must also be provided.

GENI must provide a graphical tool for browsing resources. This interface would ideally use familiar techniques used to browse resources in other domains such as searching for files in a file system or books in an on-line catalogue.

In addition to information about a resource, a GENI Rspec might have information needed by an experimenter to use the resource. The Rspec may have this information embedded in it or may have pointers to this information. GENI must provide tools for extracting this information from the Rspec and making it available to the experimenter in a useful and easy to use format.

## 5.3  Experiment Deployment

The deployment phase of a GENI experiment entails the four steps shown in Figure 6.  The *Present Credentials* step authenticates the user and verifies he is indeed permitted to lease GENI resources and use them for experimentation.

The experimenter then leases resources for the experiment.  Leases ensure requested resources are made available to the experiment at the times specified by the experimenter.  The process of leasing resources is similar to that of browsing for resources and can be done in three ways: Automatically from an experiment specification, programmatically, or using an interactive user interface.  The experimenter specifies a start time and duration for resources leases.

The experimenter can then configure the resources obtained: Install software, configure components, set topologies, turn on Internet feeds, connect to already running experiments (experiment composition), etc..

Finally, the experimenter can verify the deployment: Verify the necessary resources have been obtained and programmed with the right versions of software and traffic flows are as expected.  The verification of a deployment that uses intermittently connected resources may be a challenge.

### 5.3.1  Tools and Services for Experiment Deployment

GENI must provide researchers with a checklist of items to be considered before deploying an experiment.

GENI must provide a tool that can interpret an experiment specification and lease resources at appropriate clearinghouses for resources.

GENI must provide tools for coordinating schedules to permit sharing of resources that cannot be sliced and for composition of large experiments.  Organizations providing GENI resources might be encouraged to support a common format for calendar information.

GENI must provide a tool for managing software in distributed experiments: pushing software to large numbers of components, installing this software, tracking software versions, etc..

GENI must provide a mechanism that allows one experiment to connect with another already running experiment.

GENI must provide tools and services to validate an experiment setup: Resources allocated to the experiment, versions of software installed and communication paths set up.

An experimenter must be able to save a list of resources leased in this phase and, at a future time, request these same resources be made available to the experiment.  This will allow the experimenter to return to a configuration that has been validated.
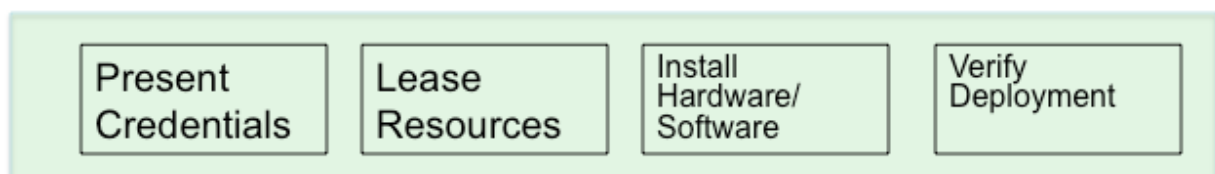


Figure 6. Steps in the Experiment Deployment phase of the experiment lifecycle.

## 5.4    Experiment Execution

Figure 7 shows the steps in the experiment execution phase of the lifecycle.  Once the experiment is set up, the experimenter controls it by starting, pausing, resuming, restarting and stopping the experiment.  Experiments can be controlled at different levels of granularity: The entire experiment, at a resource level or at an experimenter defined resource grouping.  Additionally, different experimenters may be allowed different levels of control on experiments.

Other experiment control related activities include growing or shrinking the experiment by adding or removing resources, adjusting the level of instrumentation, turning on or off flows to and from the Internet, and connecting or disconnecting from other on-going experiments.

The experimenter can turn on and off different levels of instrumentation.  Experiment instrumentation includes measurement of the performance of the experimenter's distributed software and measurement of the performance and utilization of GENI resources used by the experiment. Instrumentation data may be viewed in real-time or saved for future analysis.

The experimenter can specify automatic actions to be taken when parts of the experiment fail, if the resources being used by the experiment become unavailable or additional resources become available. The experimenter can also inject faults into the system.  Faults can include the failure of resources, a reduction in the amount of resources available to the experiment, duplicate messages, etc.

Experimenter can also monitor and maintain experiments from an operational point of view.  They can use tools to troubleshoot their experiment setup if things don't go as planned; for example, tools for browsing GENI system logs and diagnosing problems.  They may also have available a "help-desk" for help with troubleshooting experiments.

### 5.4.1    Tools and Services for Experiment Execution

GENI must provide experimenter tools to control an experiment programmatically or using an interactive tool.  An experimenter or a group of experimenters must be able to monitor and control experiments from multiple locations.  Ideally, they would be able to control an experiment from any computer using a browser-based interface.

GENI must also provide experimenters tools and libraries to implement standard distributed system mechanisms such as barriers where different parts of the distributed program cannot progress until all parts have reached the barrier; detection of distributed quiescence where the distributed program has reached a state where every program component is waiting on some other component and there are no messages in transit to and from these components; and checkpointing the state of the experiment and the resources being used.

Good instrumentation tools are critical to GENI's success because of the importance of measurement and analysis in scientific experiments.  The tools should allow the experimenter to dynamically vary the level of instrumentation, control the flow of measurements to analysis tools and

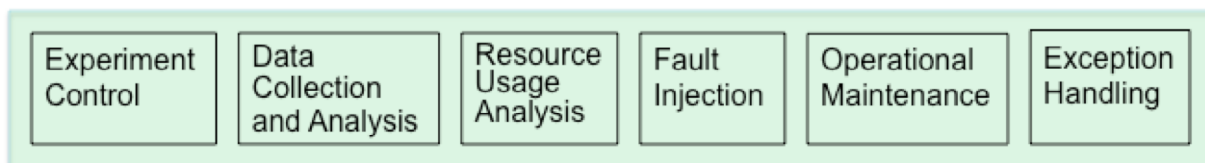| Experiment Control | Data Collection and Analysis | Resource Usage Analysis | Fault Injection | Operational Maintenance | Exception Handling |

Figure 7. Steps in the Experiment Execution Phase of the experiment lifecycle.

archive this data.  The instrumentation system must have minimal or no impact to the performance of the experiment.  As mentioned earlier, the experimenter must be able to instrument GENI resources in addition to his or her program.  Mechanisms must be provided to control access to the collected data as this some experiments may collect data or traffic that is subject to privacy policies.

GENI must provide tools to visualize the collected data.

GENI must provide tools for injecting a variety of different kinds of faults in a controlled manner. Users must be able to add their own fault-injection tools to a GENI provided fault-injection framework.

GENI must provide tools and services for debugging operational problems with an experiment such as the inability to reach or obtain certain resources or unexpected behavior of resources.  GENI may also have an help-desk for helping experimenters troubleshoot operational problems.

GENI must provide mechanisms for researchers to specify actions to be taken when certain standard or user defined exceptions occur during experimentation.

## 5.5   Experiment Sunsetting

Figure 8 shows possible options for a GENI experiment at the end of its lifecycle.  The experimenter may choose to archive the experiment so it is available to future researchers to re-run or to extend.  An experiment archive must include a formal description of the experiment in an experiment description language, a human readable description in a natural language, experimental results and any special considerations for repeating the experiment.

Technology developed by an experiment may also be transitioned to a product.  The steps to product are outside the scope of this document.

### 5.5.1   Tools and Services for Experiment Sunsetting

GENI must provide experimenters with a checklist for archiving experiments.  It must also provide facilities for archiving experiments and tools for verifying the minimum required information about an experiment is stored in the archives.

## 6   Other Experimenter Support Services

GENI researchers must have access to example programs and program outlines they can modify and extend as they develop their own distributed programs.

GENI must establish an active user group for researchers to share GENI experiences, tips and tricks and to get help with GENI from other researchers.

GENI must provide tools that make distributed, cross-institution collaboration easy.  These may include chat tools, distributed whiteboard tools and display sharing tools.

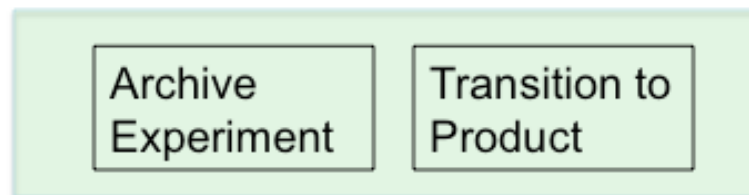All GENI tools and services must provide online help and user documentation.



Figure 8. Possible options for sunsetting a GENI experiment.

## Acronyms

The following table defines acronyms used within the GENI Project.

GENI          Global Environment for Network Innovations