

Instrumentation Tools for a GENI Prototype

Jim Griffioen and Zongming Fei
Laboratory for Advanced Networking
University of Kentucky
Lexington, KY 40506

This material is based upon work supported in part by the National Science Foundation under grant numbers DUE-0511534 and CNS-0834243. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of GPO Technologies, Corp, the GENI Project Office, or the National Science Foundation.

Project Overview

○ Goals:

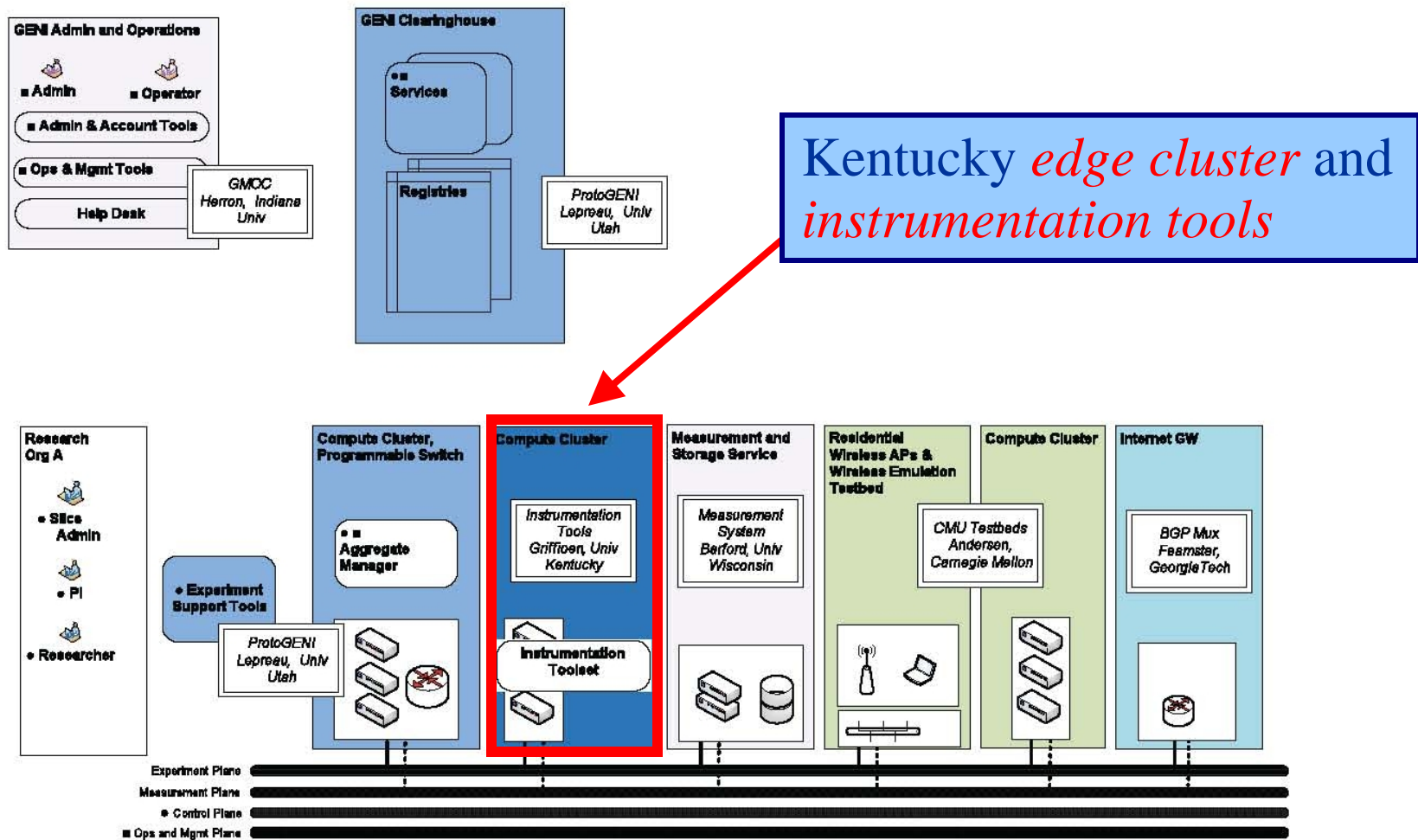
- Develop an operational GENI aggregate and integrate it into the ProtoGENI cluster.
- Create instrumentation tools that give GENI users the ability to better understand the runtime behavior of their experiments.
- Support educational use.

What do we bring to GENI?

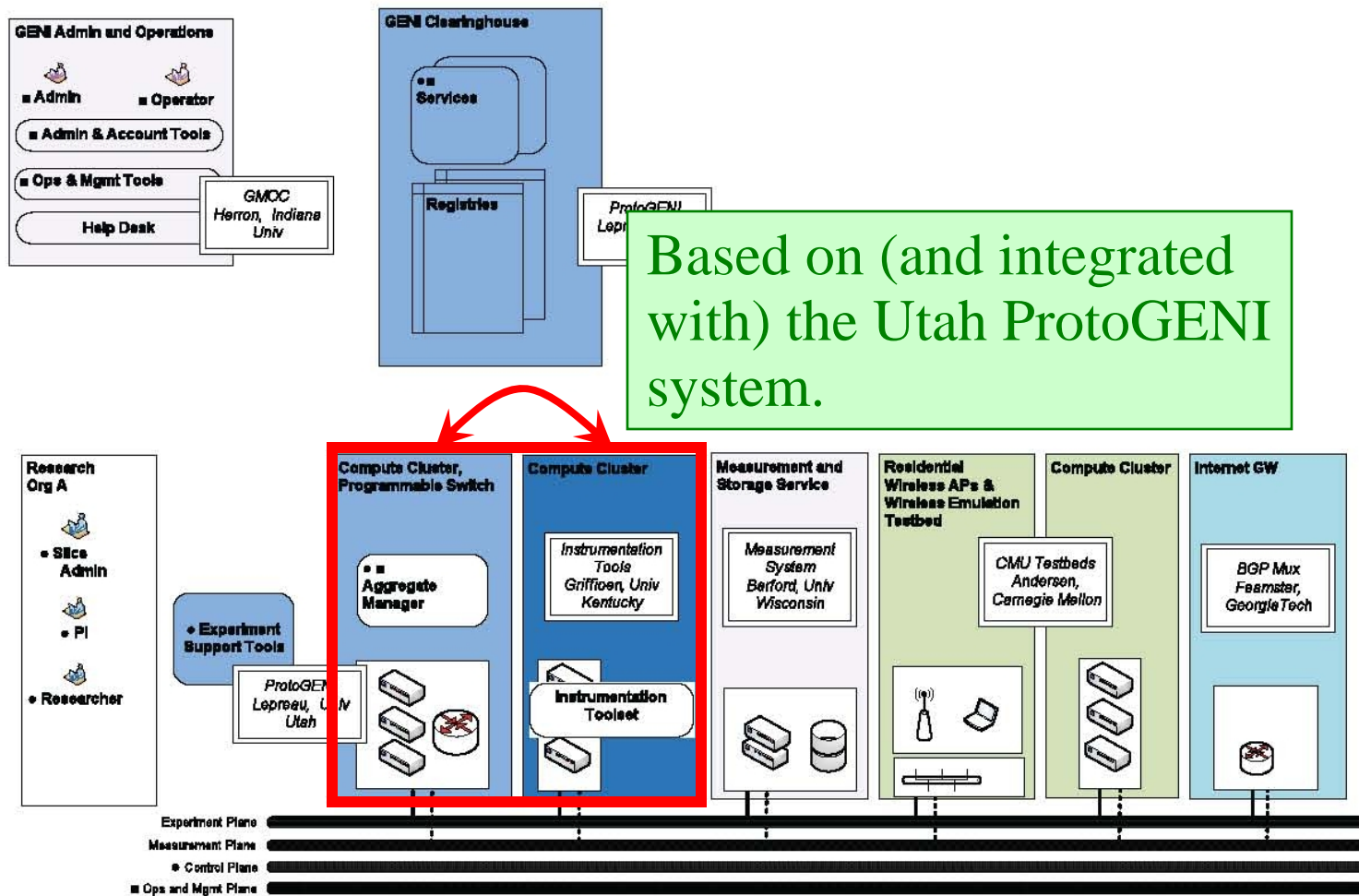
- **Kentucky Edulab** - A ~50 node Emulab-based system designed for education as well as research.
- **Operational Experience** - many years operating an Emulab facility.
- **User Community** - consisting of researchers, students, faculty, TAs, etc.
- **Enhanced Instrumentation Tools** that make it easier for users to monitor and understand the (runtime) behavior of their experiment.



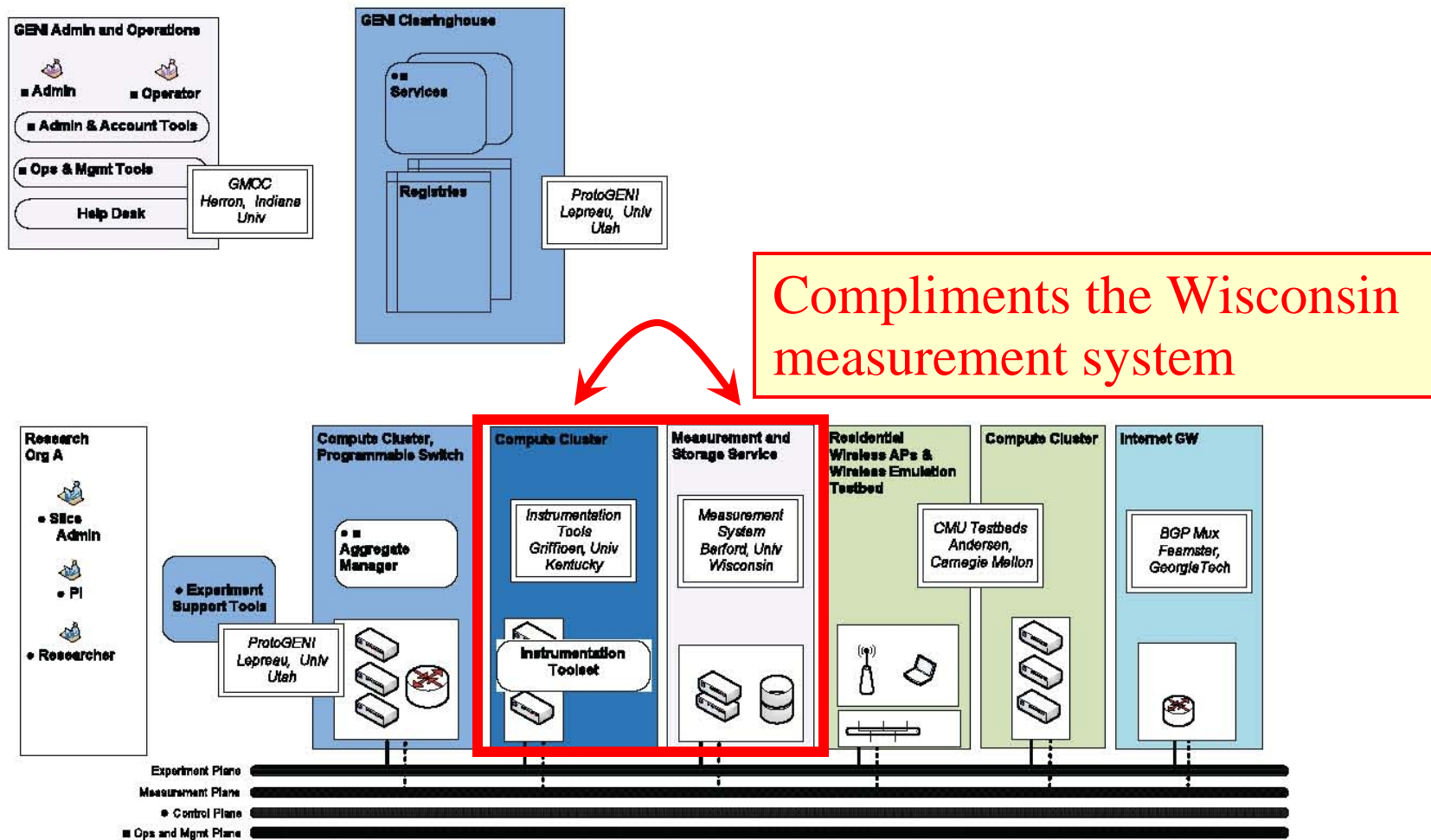
Cluster C: ProtoGENI



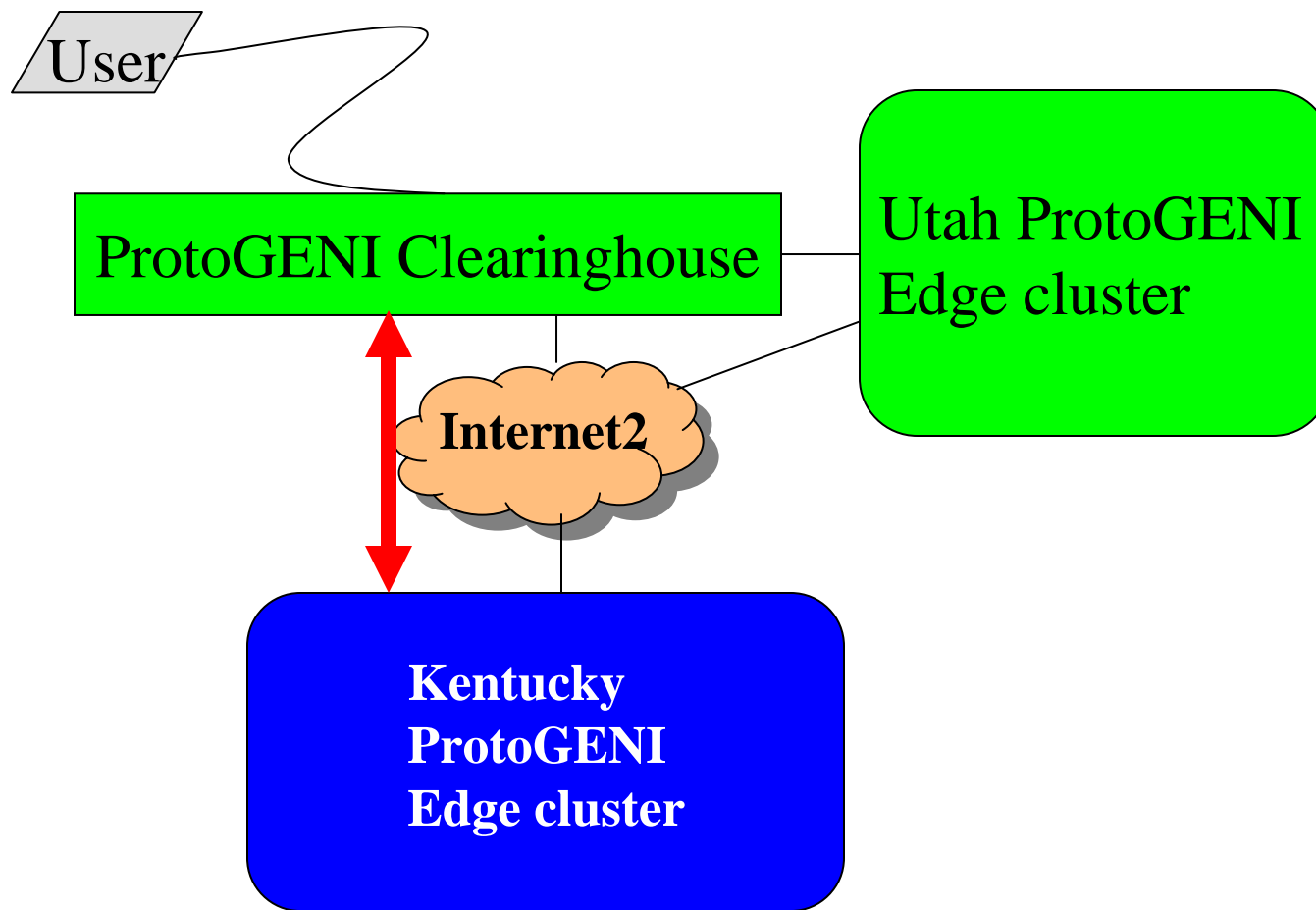
Cluster C: ProtoGENI



Cluster C: ProtoGENI



Relationship and Connectivity to ProtoGENI



Workplan

- Modify/upgrade the existing Edulab hardware.
- Install the ProtoGENI code base and create an operational aggregate.
- Integrate our Edulab aggregate with the ProtoGENI clearing house.
- Federate with other ProtoGENI aggregates in our cluster.
- Port the Edulab instrumentation tools to the ProtoGENI environment and enhance with new features/tools.

The Importance of Monitoring

- Much of the work on testbeds has focused on creating, setting up, and running an experiment; but this is only the first of many steps in an experiment.
- The real challenge can often be monitoring and analyzing the behavior of an experiment; it can be a very involved, time consuming, manual process that is repeated many times.
- It involves setting up and launching a monitoring environment.

Monitoring Challenges

- Multiple uses
 - Debug
 - ◆ Verify the configuration
 - ◆ Verify the experiment is behaving correctly
 - Learn/Discover
 - ◆ Understand what happens and why
 - ◆ Measure performance
- Requires
 - deciding what information should be monitored about the network, OS, and applications
 - selecting the level of detail for monitoring
 - monitoring desired information
 - filtering information to be reported
 - collecting/transporting recorded data
 - making information available/viewable to users.

Monitoring Challenges (2)

- Scaling to large experiments.
 - Monitoring load on components
 - Collect/transport instrumentation data
 - ◆ Without interfering with the experiment
 - ◆ Collecting data from many components
- Logging for history view and archival purposes.
- Security and authorization to access monitoring capabilities.
- Sandboxing the monitoring system.
- Accounting for monitoring activity.

Existing Solutions

- Software and tools exist, but they
 - often take an expert to use
 - can be cumbersome to use
 - do not necessarily scale
 - are rarely non-intrusive
 - often use non-standard formats/interfaces

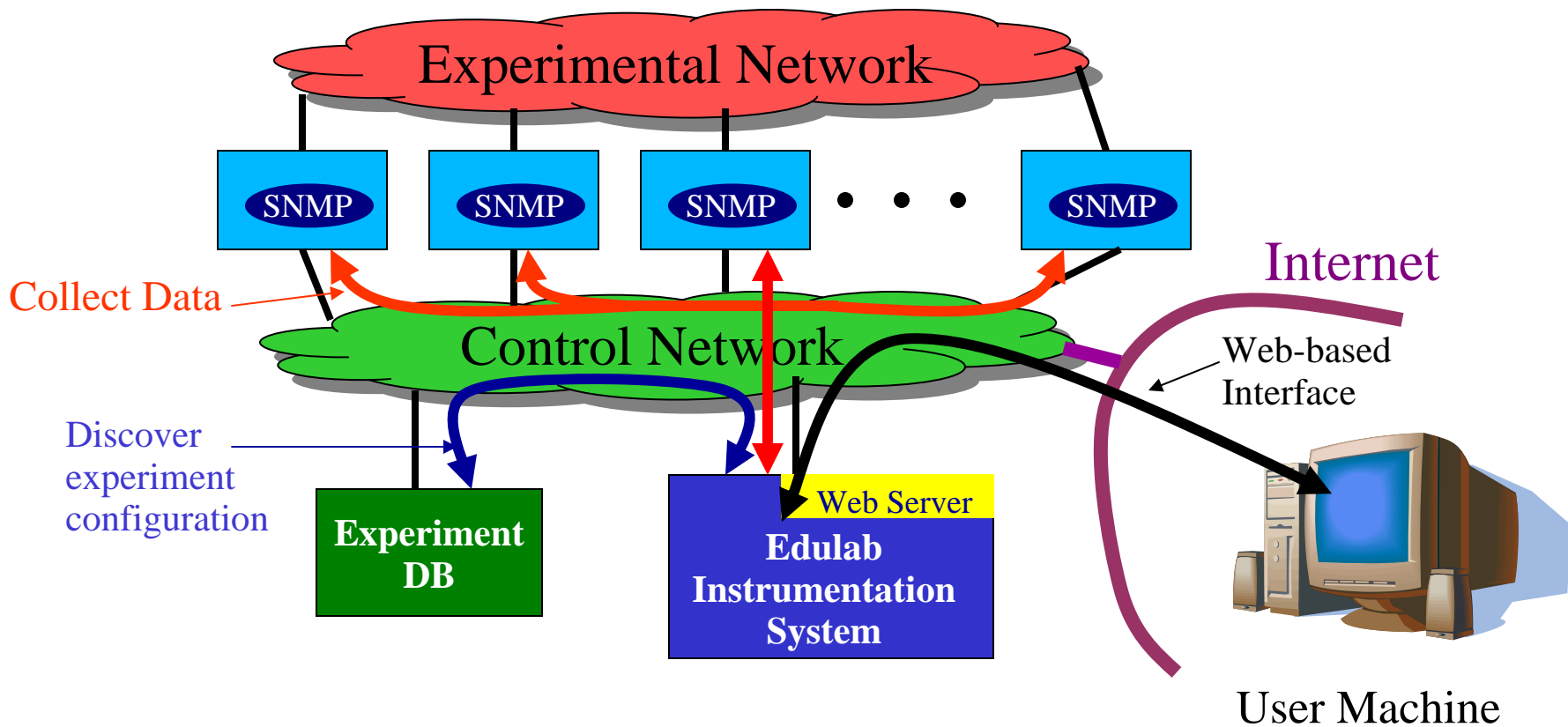
Educational Requirements

- Users are students who
 - are unfamiliar with the testbed
 - do not want to become testbed experts
 - need to get up to speed fast
 - are not experts in system administration (configuration, monitoring, debugging)
 - often do not care about the low-level details of the environment (hardware/software). They want to think about logical components.
 - have limited screen space.
 - find that pictures/GUIs are worth a 1000 words
- Users are also faculty, TAs, lab assistants, and graders each with slightly different usage needs.
- Experiments are shared by multiple users.
- Need to exchange monitoring information.
- Virtualization is important for scalability
- Long term and short term scheduling can also improve scalability

EduLab Approach

- Make it easy for users to monitor their (running) experiment.
- Leverage existing monitoring capabilities
- Use pictures and GUI's to simplify the learning curve.
- Select options on user's behalf
- Support drill-down
 - Packet-level details obtained on request
 - (would like) connection-level info

Edulab Architecture



Edulab Instrumentation Tools

- Modified the Emulab web interface to include experiment monitoring pages
- Users select link or node of interest to observe characteristics/performance
- Reports information about
 - OS configuration
 - OS performance/load
 - Traffic graphs
 - Packet traces

Edulab Instructional Features

- Automated user registration
- Experiment snapshots
- Project scheduling
- Support for virtual nodes
 - Focused on measurement load imposed on nodes.
 - Ignored network cost because we utilized the Emulab control net - this will need to change to the GENI measurement plane

Thank You!

Questions?