

The Usage of a Programmable Edge Node in ProtoGENI

University of Massachusetts Lowell

September 2009

Abstract

A high performance and flexible programmable edge node (PEN) is an important element to the GENI sites federating into the GENI global network and to the users carrying out experiments. The PENs provide virtual network interface cards (NICs) and virtual containers to support concurrent GENI experiments on the shared substrate. We have developed a PEN prototype based on multi-core x86 processors and network processors (NPs) and integrated with ProtoGENI control framework. In this document, we describe a usage example of how users can run experiments on PEN through ProtoGENI's Flash interface.

1. Introduction

We have developed a Programmable Edge Node prototype with general-purpose and application-specific processors (network processors.) for providing virtual containers and isolating experiments from measurements. We have also integrated the PEN prototype with ProtoGENI control framework to facilitate the usage of the resources on the PEN.

In this document, we describe a usage example of creating an experiment on PEN with ProtoGENI's Flash interface. We assume the users have obtained credentials to access ProtoGENI's Flash interface and have reasonable knowledge of the network experimentation with Emulab/ProtoGENI.

This document is a live document that we are constantly updating with new usage examples.

2. Usage Example

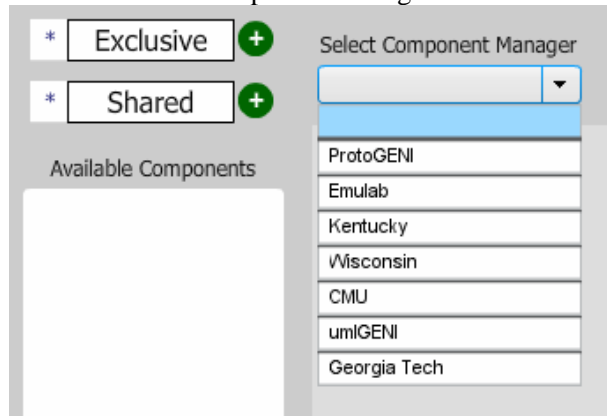
- User opens the Flash Interface
 - Interface requests a Slice Name



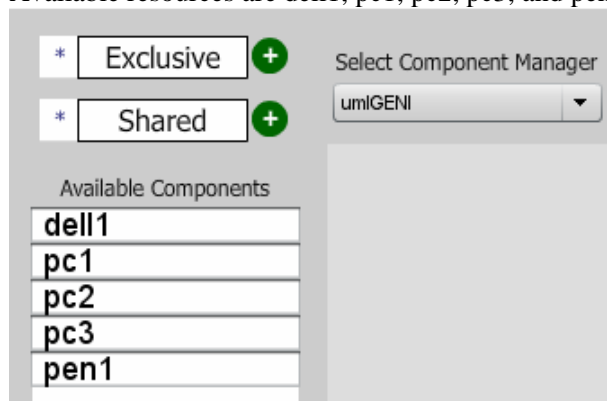
The screenshot shows the ProtoGENI Flash interface. At the top, the word "ProtoGENI" is displayed in a stylized font. Below this, there are two input fields. The first is labeled "Slice Authority:" and contains the text "urn:publicid:IDN+emulab.net+authority+sa". The second is labeled "Slice Name:" and contains the text "demoslice". Below these fields is a blue button labeled "Create Slice".

- User clicks 'Create Slice'
 - Interface checks for valid credentials

- Interface creates a new Slice
- Flash Interface queries all known component managers for available resources
 - List of available component managers becomes accessible to the user

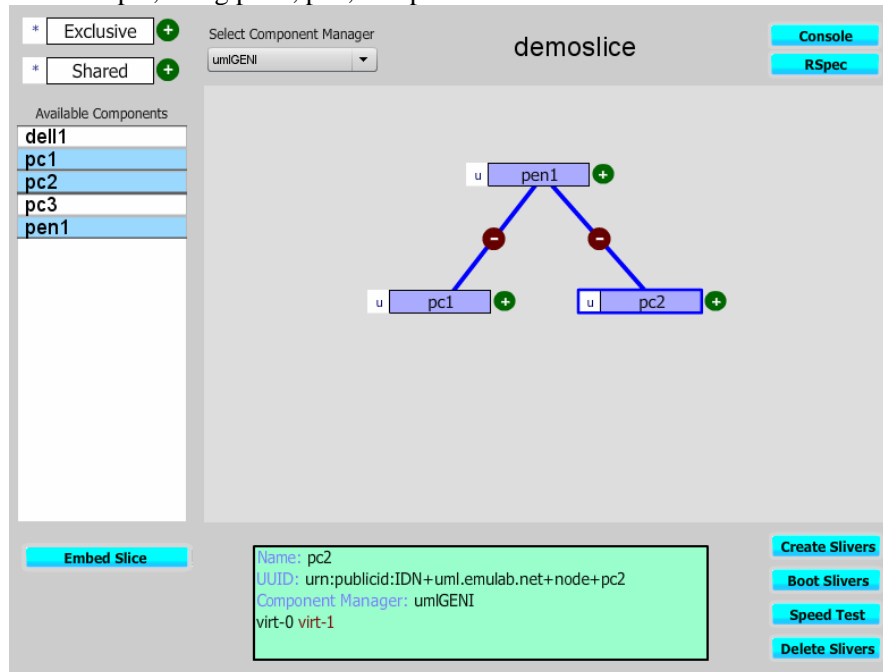


- User selects the umlGENI component manager
 - Available resources are dell1, pc1, pc2, pc3, and pen1



- User creates a desired topology

- For example, using pen1, pc1, and pc2



- User selects 'Create Slivers'
 - Flash Interface sends an RSpec to the UML Component Manager which details the desired topology
 - The UML Component Manager allocates resources to satisfy the request
- User selects 'Boot Sliver'
 - Flash Interface sends the request to the UML Component Manager
 - The UML Component Manager boots regular nodes and ignores the PEN node
 - i. The PEN node does not need to be booted because it became live during the 'Create Slivers' phase
- User is now able to log into and use the allocated nodes
- User selects 'Delete Slivers'
 - The UML Component Manager releases and reboots physical nodes
 - The UML Component Manager calls custom scripts to bring down and release the virtual node on PEN
- Process is now complete

3. Conclusion

We have integrated our PEN prototype with ProtoGENI control framework. Users are able to request resources from the PEN-type nodes and deploy experiments. We are in the progress of further testing the operation of the prototype and creating more usage examples. We welcome early trial usage and comments from interested users and GENI developers.

4. Further Readings

For information about the UMLPEN project, please visit <http://cans.uml.edu/index.php?Research.PEN>

For information about ProtoGENI, please visit <http://www.protogeni.net/> .

For information about the design of the PEN prototype, please refer to the design document titled “*The Design of a Programmable Edge Node for GENI*” available at the project website <http://cans.uml.edu/index.php?n=Research.PEN> .

For information about how the PEN fits in GENI, please refer to the document entitled “*The Integration of Programmable Edge Node with ProtoGENI Control Framework*”, available at the project website.