

VIRO-GENI: Deployment of a plug & play, scalable, robust virtual Id routing in GENI



Braulio Dumba, Guobao Sun, Hesham Mekky, Zhi-Li Zhang
Department of Computer Science, University of Minnesota-Twin Cities, MN, 55455



This project is supported in part by a Raytheon/NSF subcontract 9500012169/CNS-134688, the NSF grants CNS-1017092, CNS-1117536 and CRI-1305237, and the DTRA grant HDTRA1-09-1-0050.

MOTIVATION

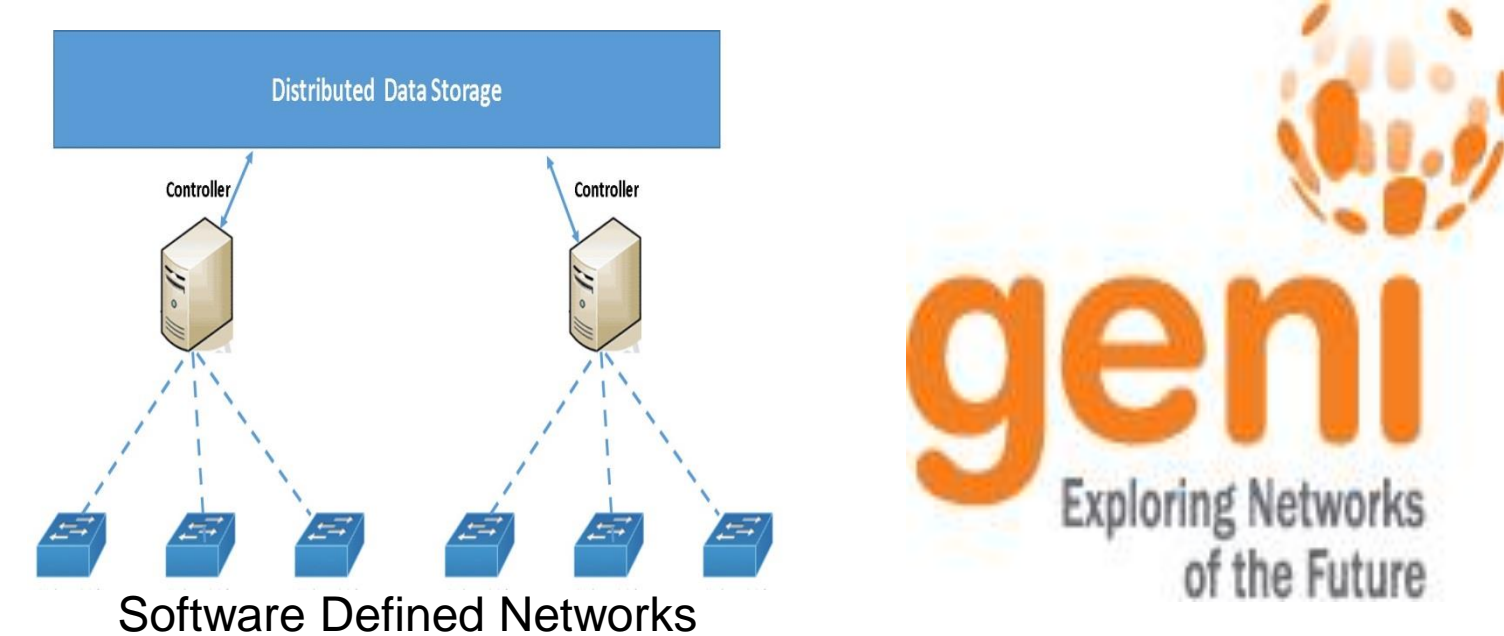
IP Known limitations:

- Poor support for mobility
- Need of careful and extensive network configurations

VIRO-GENI project goal: to implement VIRO as a non-IP service in GENI using the SDN platform

VIRO: three main components

- Virtual Id space construction
- Routing tables computation
- Data forwarding using virtual Ids

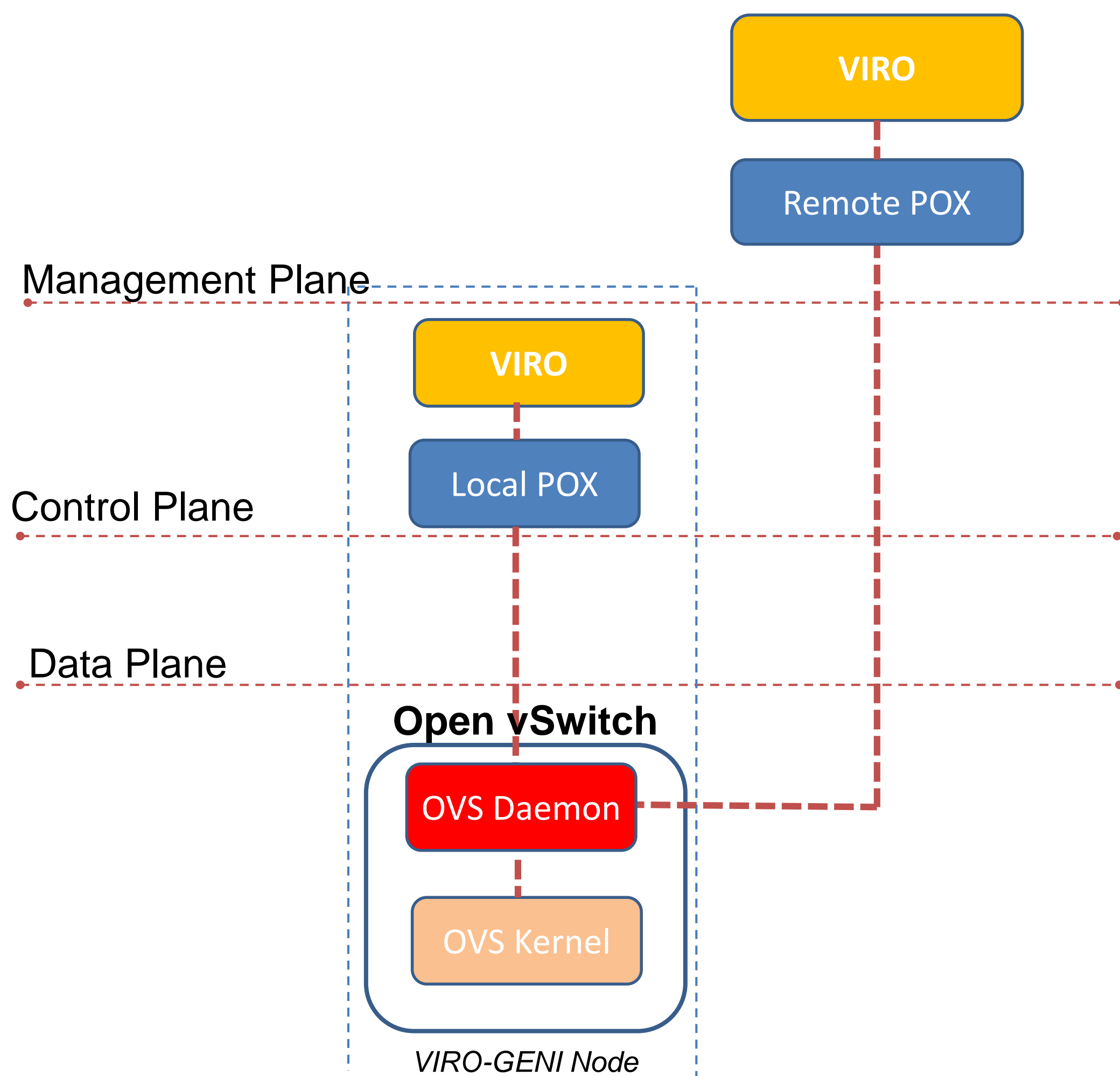


CHALLENGES:

- OpenFlow protocol flexible forwarding behavior is still tied to the standard Ethernet/IP/TCP protocol task
- VIRO has its own "topology-aware" addressing and forwarding behavior, where forwarding is based on the destination vid and the forwarding directive

IMPLEMENTATION OF VIRO IN GENI

VIRO-GENI NODE



CONTROL PLANES

Management Plane: VIRO remote controller is responsible for the following tasks:

- topology discovery/maintenance (host/switch added/removed)
- Vid assignment
- ARP and DHCP Requests
- IP/VID Mapping (Global View)

Control Plane: VIRO local controllers are responsible for the following tasks:

- MAC/VID Mapping (Local View)
- Populate Routing Table
- Insert forwarding rules for the first packet of any flow

DATA PLANE

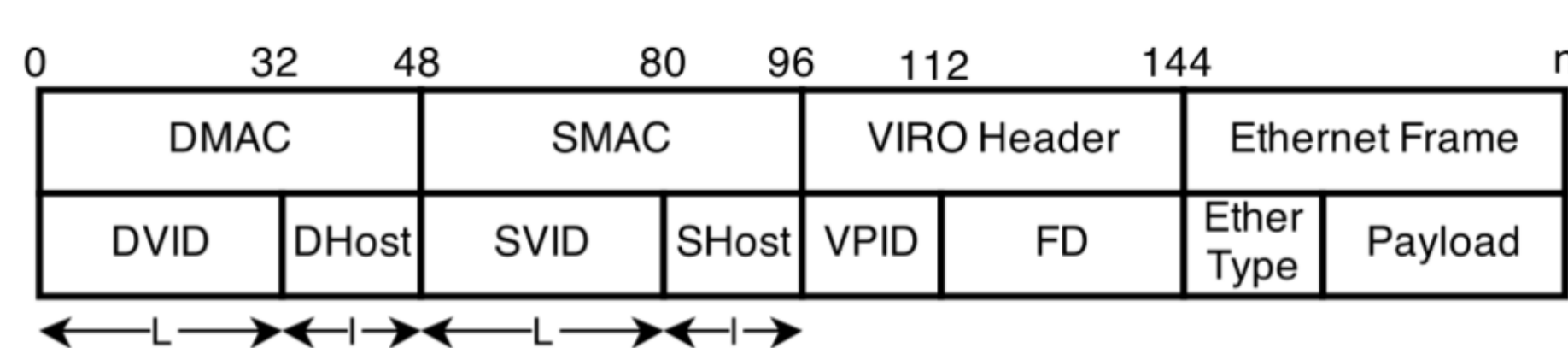
OVS Daemon:

- Translation between IP packets/VIRO packets (EtherType, Forwarding Directive)
- Insert rules for routing at Kernel

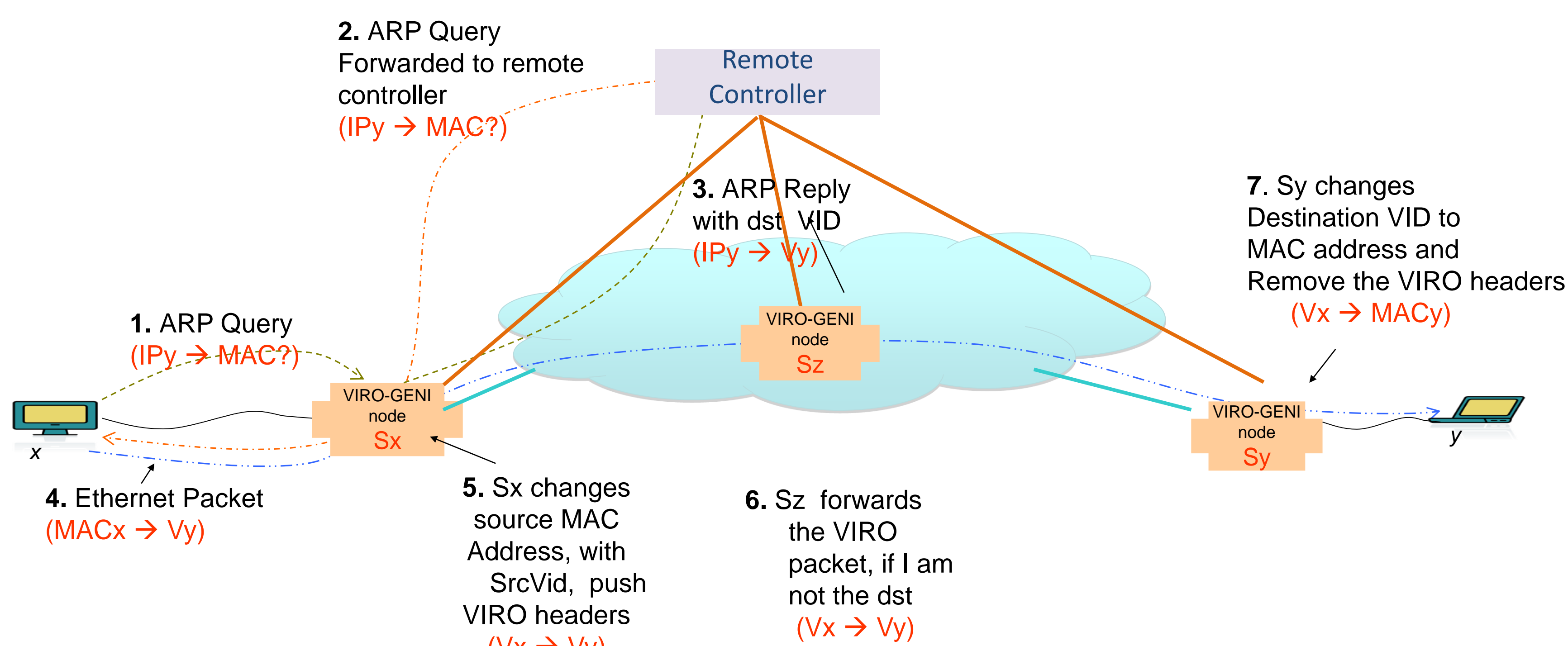
OVS Kernel:

- Translation between IP packets/VIRO packets (End-Host)
- Forwarding IP packets among local machines
- Forwarding VIRO packets

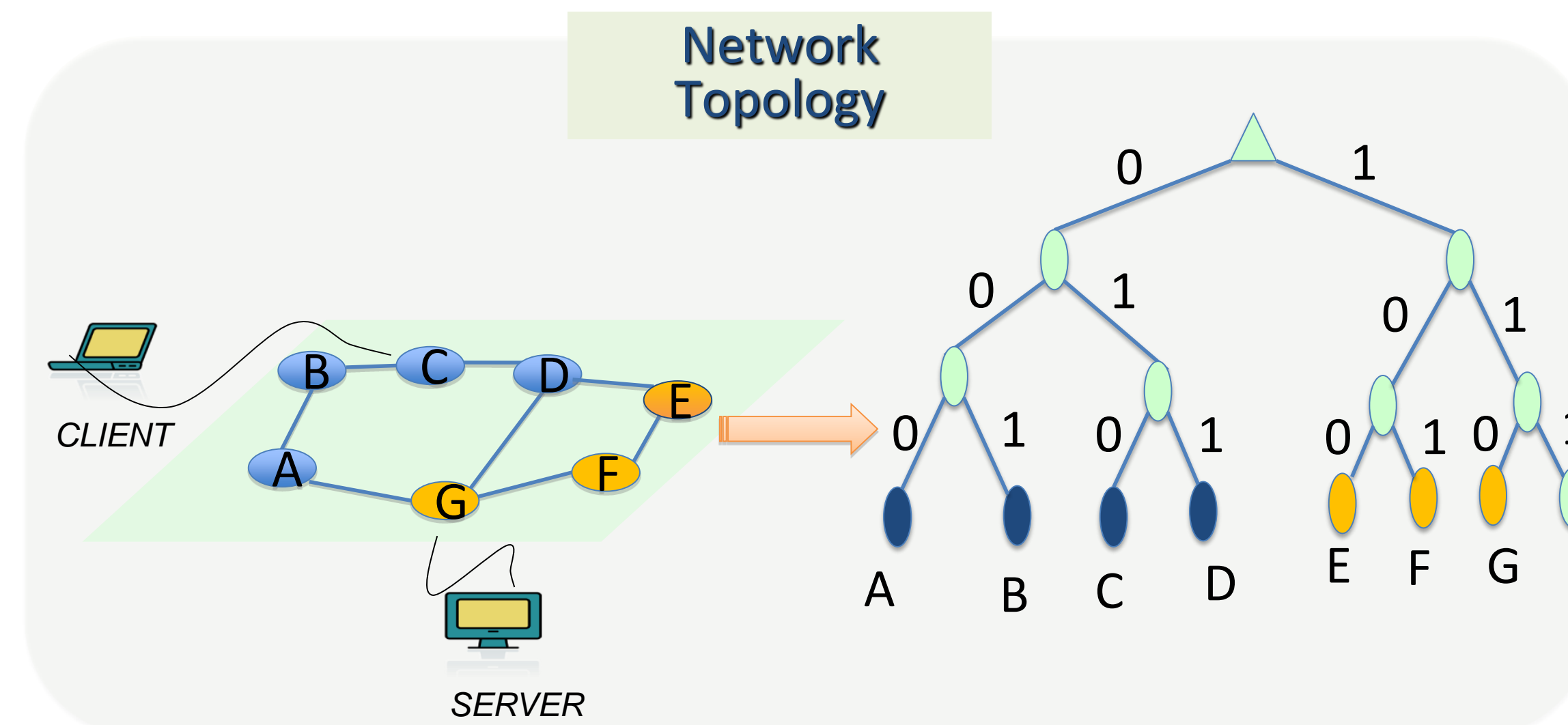
VIRO PACKET FORMAT



ADDRESS /VID RESOLUTION AND DATA FORWARDING IN VIRO-GENI



DEMO OF INITIAL PROTOTYPE

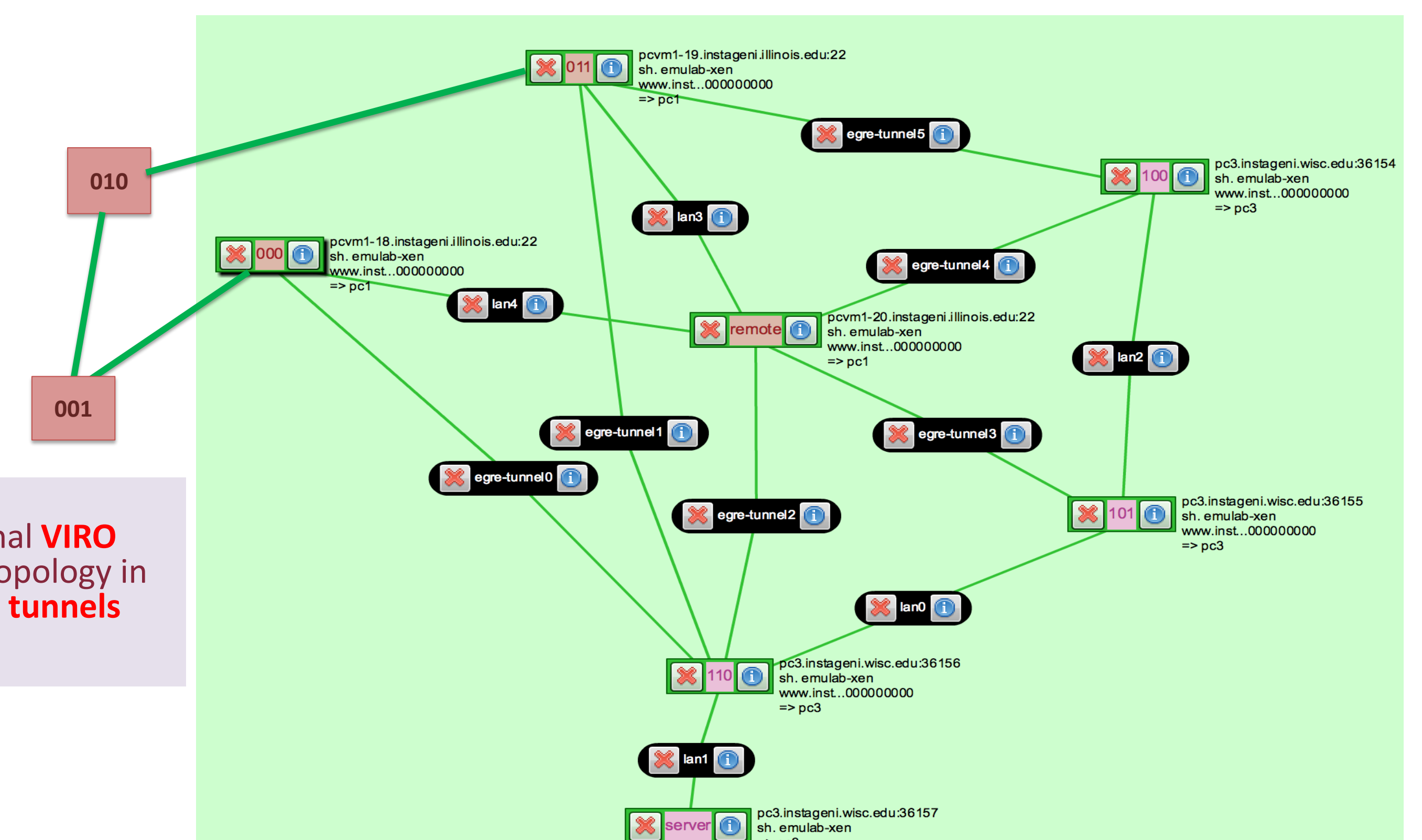


- Our goal is to show how VIRO handles host mobility.
- To pursue this goal, we carried out experiments in GENI using a network topology with 7 nodes

- The leaf nodes in the binary tree represent VIRO switches
- The color of the nodes represent GENI Aggregate Managers (AM)
- We attached a client host at node C and an Apache server at node G

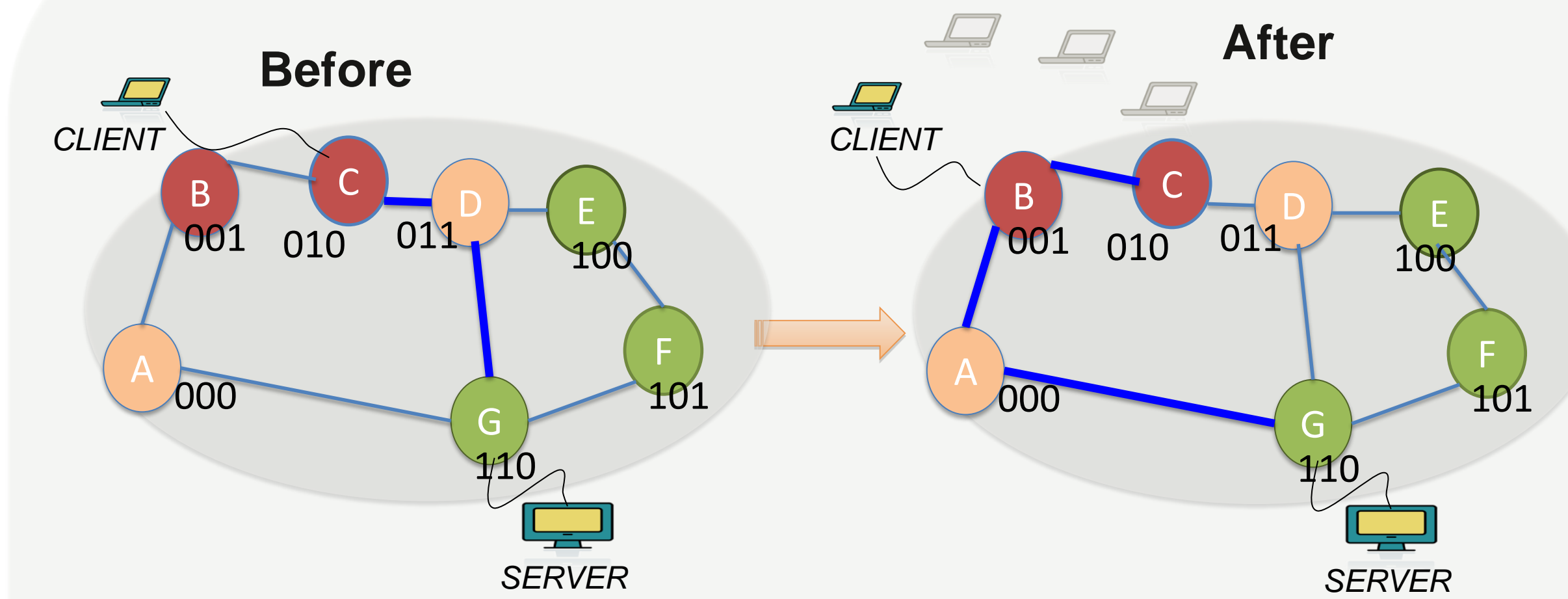
DEMO SET-UP IN GENI

- We use two GENI AMs (Wisconsin and Illinois), 7 XenVMs and 4 PCs in our experiment. EGRE tunnels were used to connect nodes at different GENI AMs.



- We connect 2 external VIRO nodes to our VIRO topology in GENI using Ethernet tunnels

Host Mobility Experiment



Bucket Distance	Next hop	Gateway
1	D	C
2	B	C
3	D	D

Routing Table for node C (Round 3)

Bucket Distance	Next hop	Gateway
1	A	B
2	C	B
3	A	A

Routing Table for node B (Round 3)

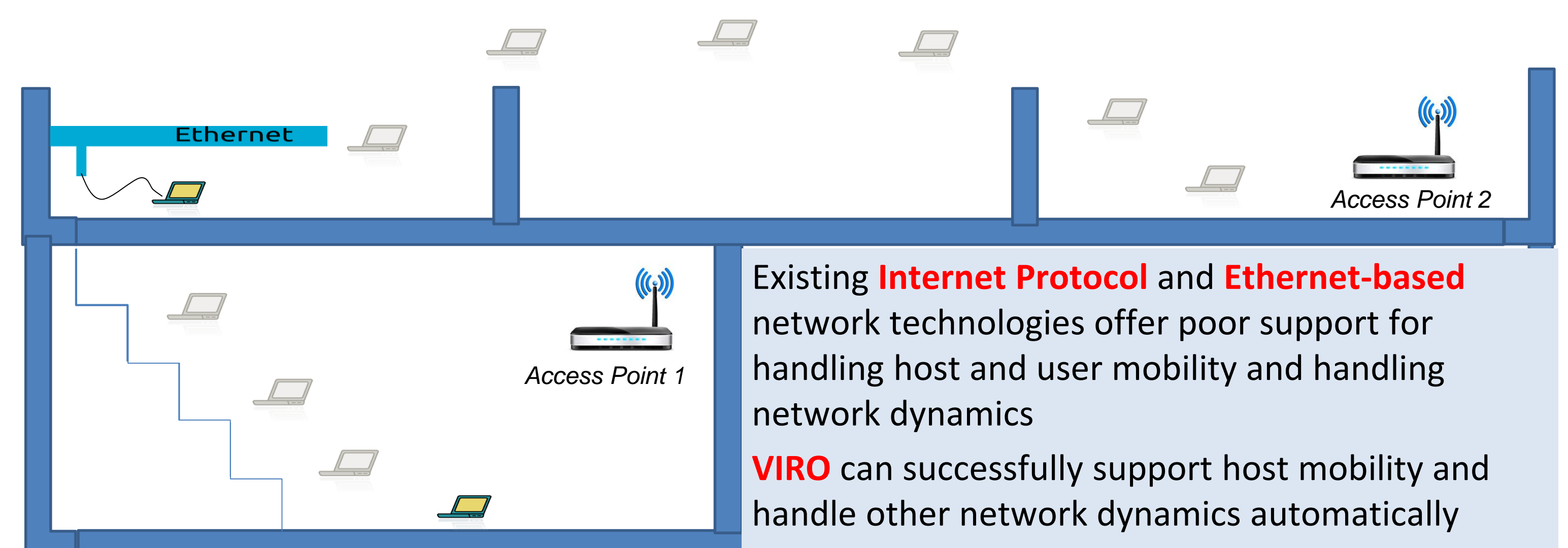
In this experiment the client at node C downloads a large image from the server at node G

Before: Node C uses its level-3 gateway (node D) to communicate with the server

After:

- The client is assigned a new vid, after moving to node B
- The server issues an ARP request to the remote controller to find the client new vid
- The client TCP connection is unaffected during this process

VIRO topology-aware, structure virtual id (vid) space offers support for host mobility



Existing Internet Protocol and Ethernet-based network technologies offer poor support for handling host and user mobility and handling network dynamics

VIRO can successfully support host mobility and handle other network dynamics automatically