Shakedown Experimentations and Prototype Services on Scalable, Agile, Robust, and Secure Multi-Domain Software Defined Networks

Lei Liu, Roberto Proietti, Xiaotao Feng, Matt Bishop, Chen-Nee Chuah, S.J. B Yoo (PI)

Department of Electrical and Computer Engineering, University of California, Davis, California 95616

Abstract

This project pursues design, operation, and experimentations on multi-domain software-defined-networks. We demonstrate inter-domain OpenFlow controllers.

Project Objectives

Cloud of Software Defined Network Control & Management Systems in multiple administrative domains

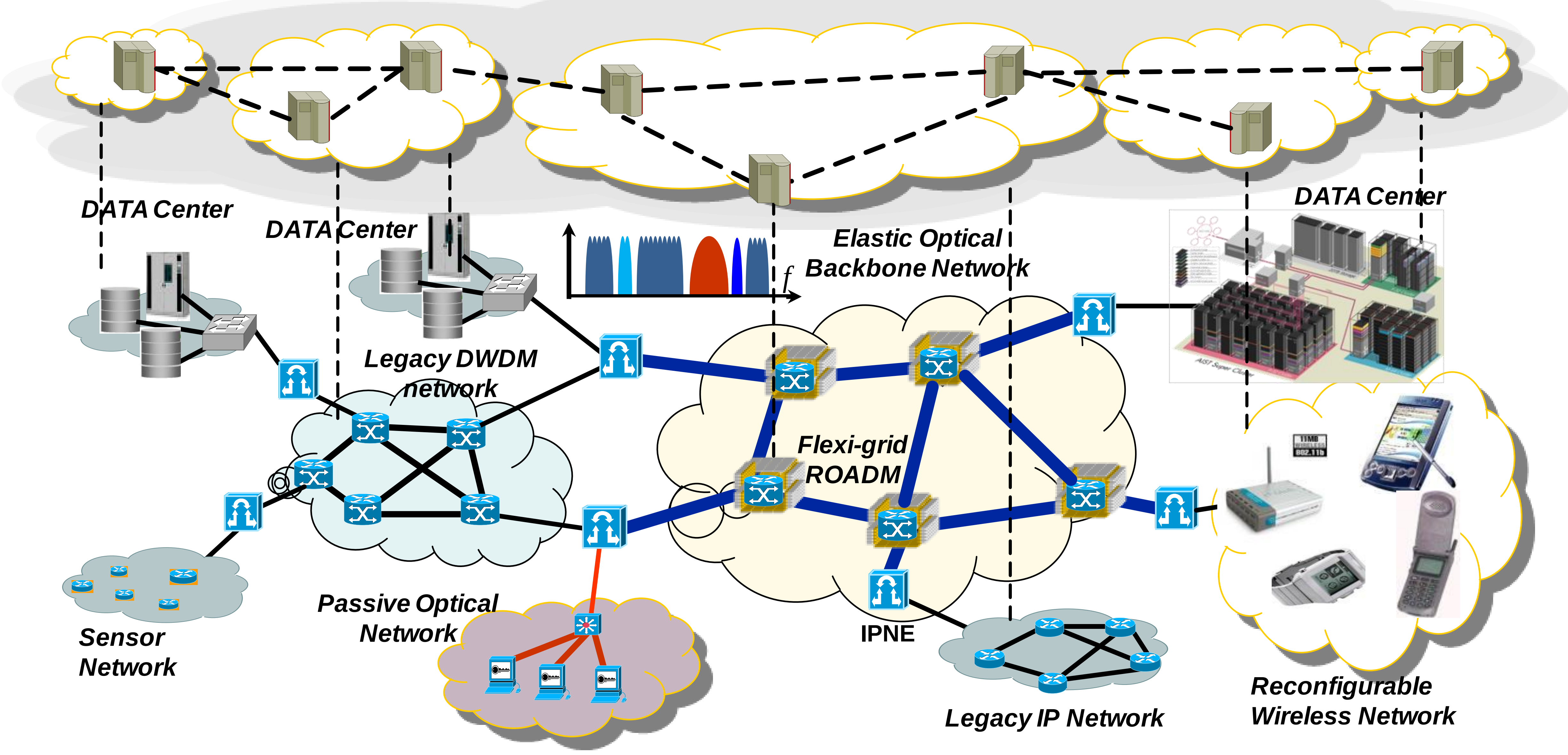


Figure 1. Multi-domain software-defined heterogeneous networking

- **Conduct a number of key experiments testing a new cross-layer and multi-domain OpenFlow control and management mechanism across multiple GENI testbeds while paying special attentions to configurability, security, and monitoring**

UC Davis Research Network

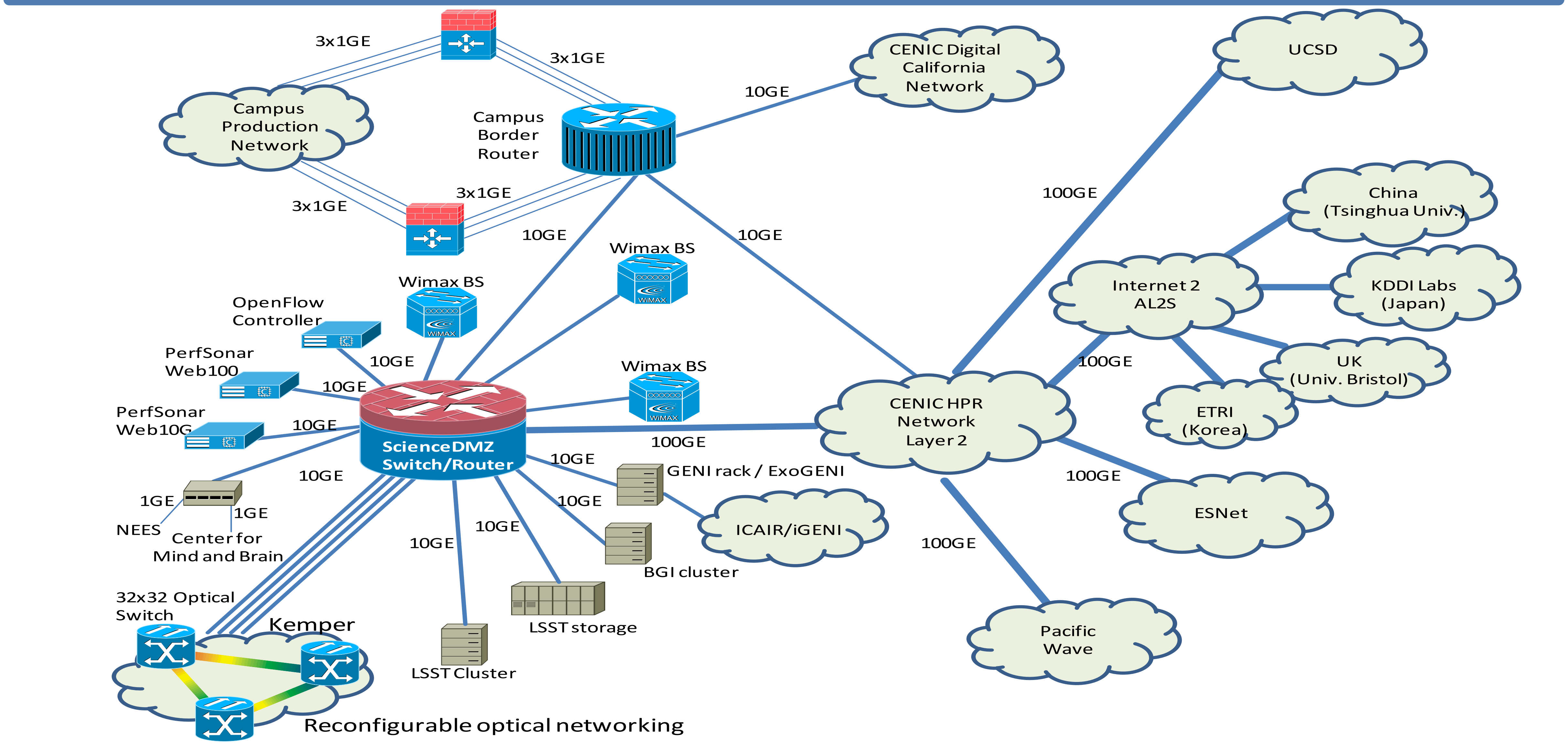


Figure 2. UC Davis campus network and its connection to other testbeds

OpenFlow controllers for inter-domain networks (demonstration)

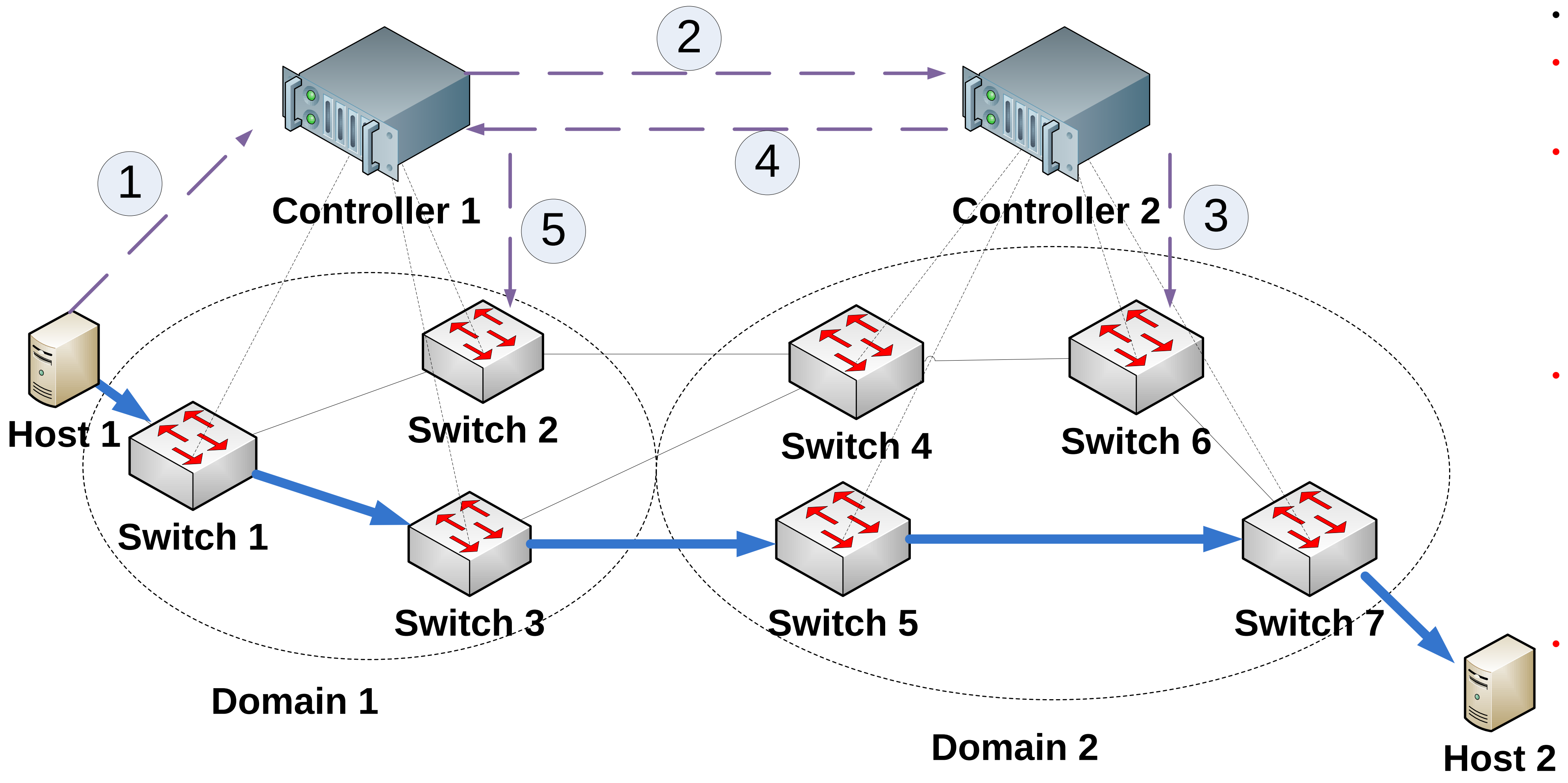
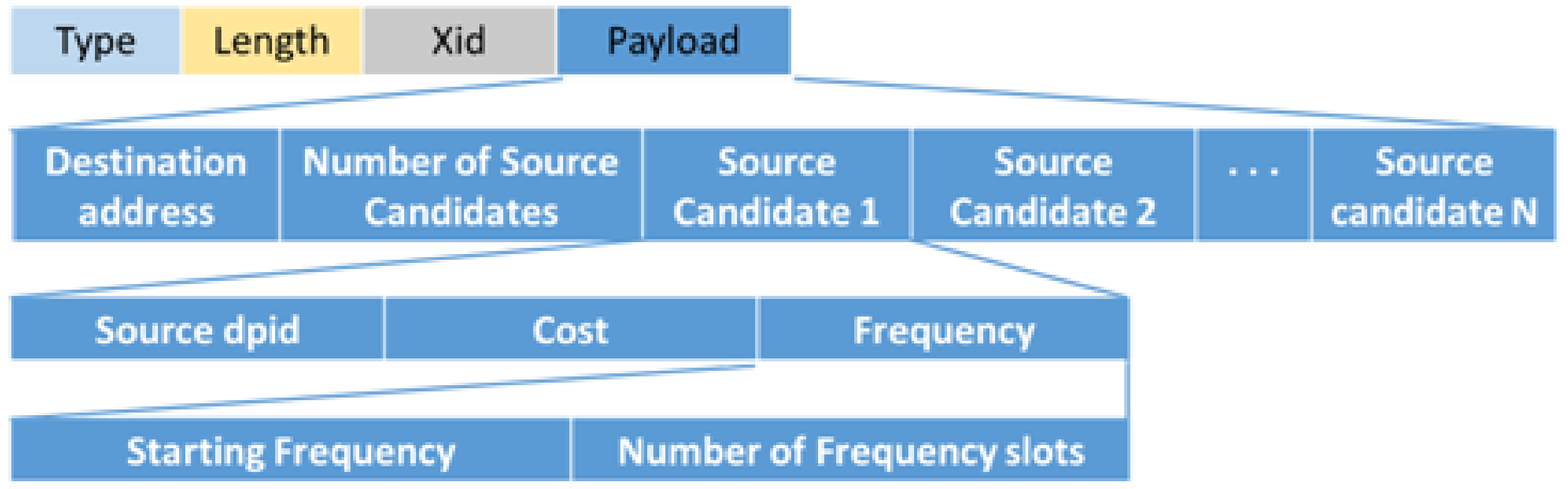


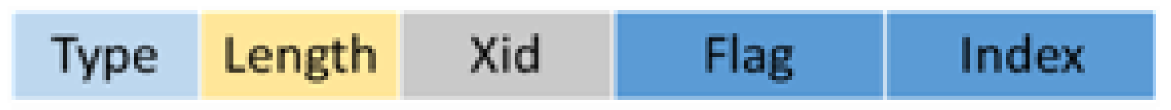
Figure 3. Demo scenario

- Procedures:
  - Step 1: An inter-domain flow request arrives at OpenFlow controller 1
  - Step 2: OpenFlow controller 1 calculates paths from Source to all domain egresses in Domain 1, and then, sends out a Path Computation Request (PCReq) to OpenFlow controller 2
  - Step 3: OpenFlow controller 2 calculates paths from ingresses to Destination in Domain 2; Combines the paths calculated in Domain 1, and finds the best end-to-end path. After that, OpenFlow controller 2 configures the switches in domain 2.
  - Step 4: OpenFlow controller 2 sends the Path Computation Reply (PCRep) message to controller 1, indicating the selected path in domain 1.
  - Step 5: OpenFlow controller 1 configures the switches in Domain 1.

Packet format of the PCRep message



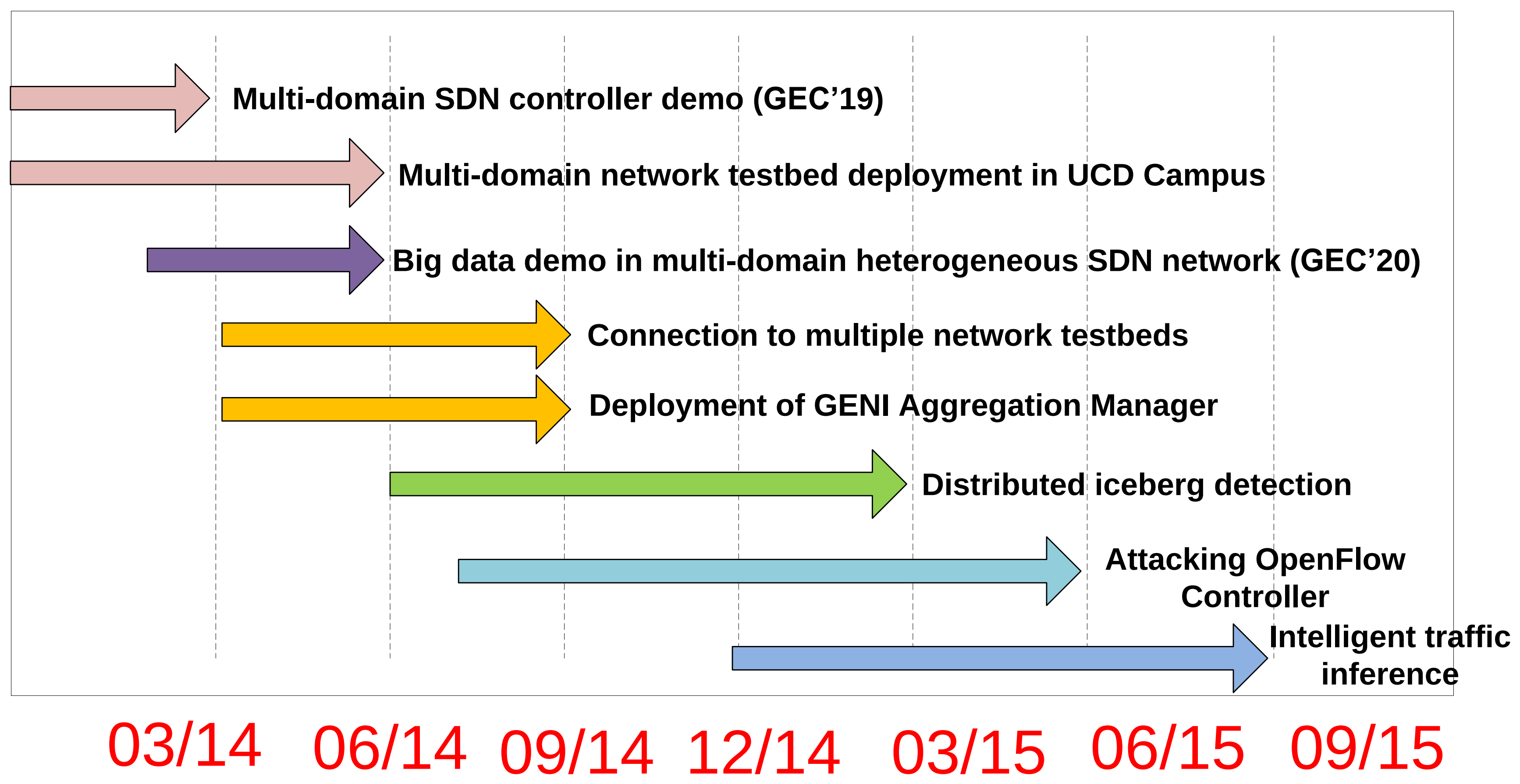
Packet format of the PCReq message



- Type (8 bits): indicate the message type
- Length (8 bits): indicate the total message length
- Xid (8 bits): indicate the message id
- Destination address (32 bits)
- Source Candidate (32 bits): indicate the candidate paths in the upstream domain
- Cost (8 bits): cost of a calculated path (e.g. hop)
- Starting frequency (32 bits): wavelength information if an optical networking is deployed
- Number of Frequency slots (32 bits): bandwidth information if an elastic optical networking is deployed
- Flag (1 bit): indicate the successful path computation in the downstream domain

- Dynamic end-to-end path computation and provisioning by collaborations of multiple OpenFlow controllers
- Avoid using a powerful parent controller on top of child controllers in each domain

Roadmap



Our demo in GEC'20 in UC Davis Campus and GENI testbed

Application aware Big Data experiment across multi-domain, heterogeneous network so that the nature of the Big Data applications should utilize the best resources.

sbyoo@ucdavis.edu