ComputerScience

RINA A Better Network Structure

BOSTON UNIVERSITY

Video Streaming over ProtoRINA

Yuefeng Wang   Nabeel Akhtar   Ibrahim Matta

Boston University

What is RINA? [1][2][3]

- RINA: Recursive InterNetwork Architecture
- A clean-slate network architecture which overcomes inherent weaknesses of the current Internet, e.g. security, mobility support
- Based on the fundamental principle that *networking is Inter-Process Communication (IPC) and only IPC*
- Distributed IPC Facility (DIF): a collection of distributed IPC processes with shared states. They provide communication service to application processes over a certain scope (i.e., range of operation)
- Distributed Application Facility (DAF): a set of application processes cooperating to perform a certain function. The function can be a communication service, weather forecast, genomics, *etc.*
- Two design principles: (i) divide and conquer (recursion), and (ii) separation of mechanisms and policies

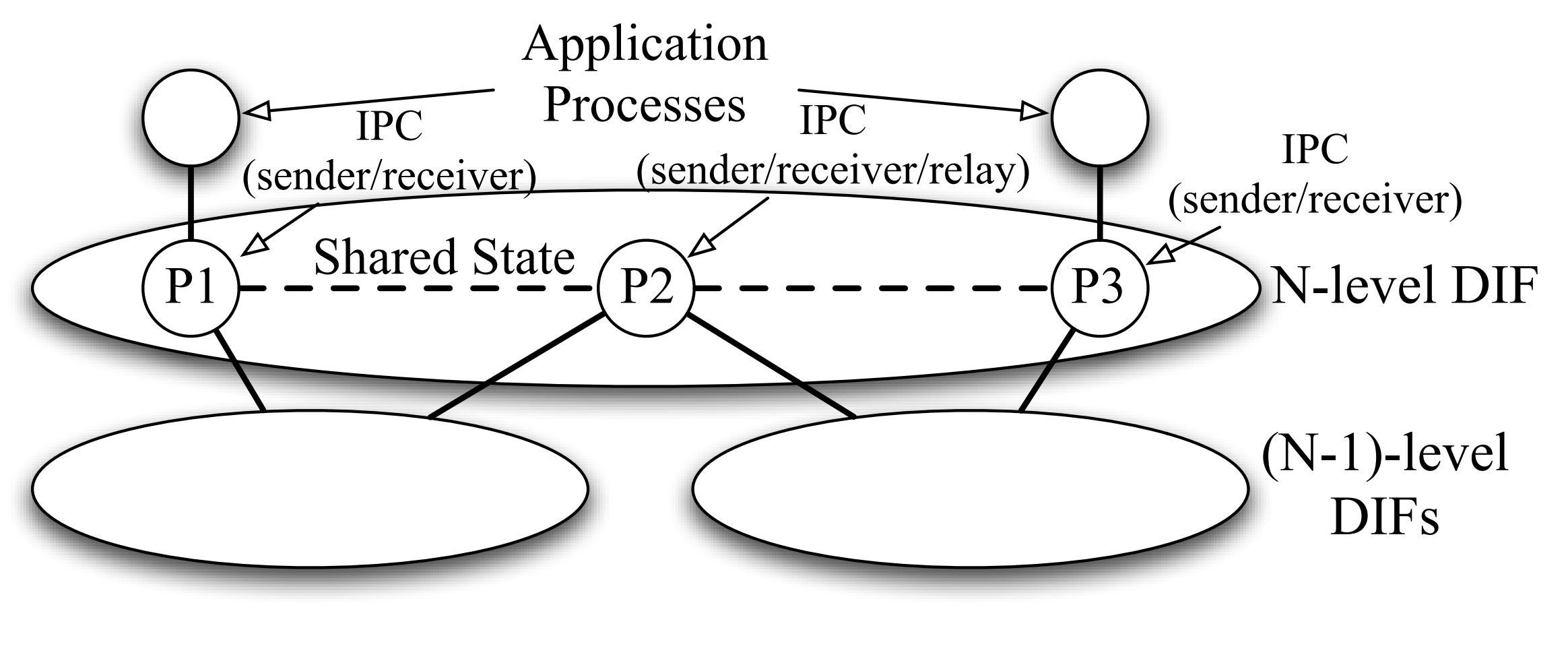

Fig 1 : RINA overview

RINA Prototype: ProtoRINA [4][5]

- ProtoRINA is Boston University's user-space prototype of RINA
- A prototype enables the programming of recursive-networking policies
- Experimental tool for developing (non-IP based) user and management applications
- Teaching tool for networking and distributed systems classes
- Version 1.0 released on October 2013; around 55,000 lines of Java code following the RINA specifications of January 2013
- Disclaimer: The current version is not a complete implementation of RINA and we continue to modify and add elements

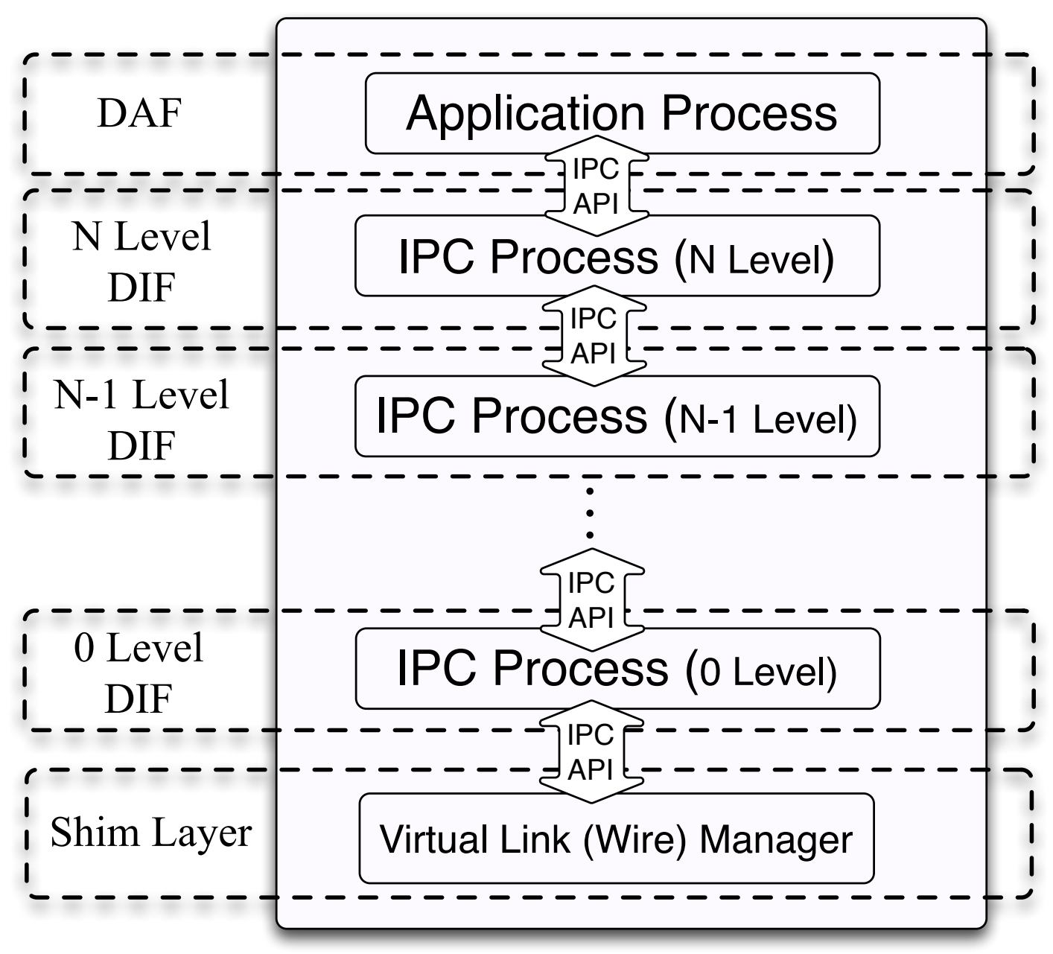

Fig 2 : RINA node is a host where processes reside

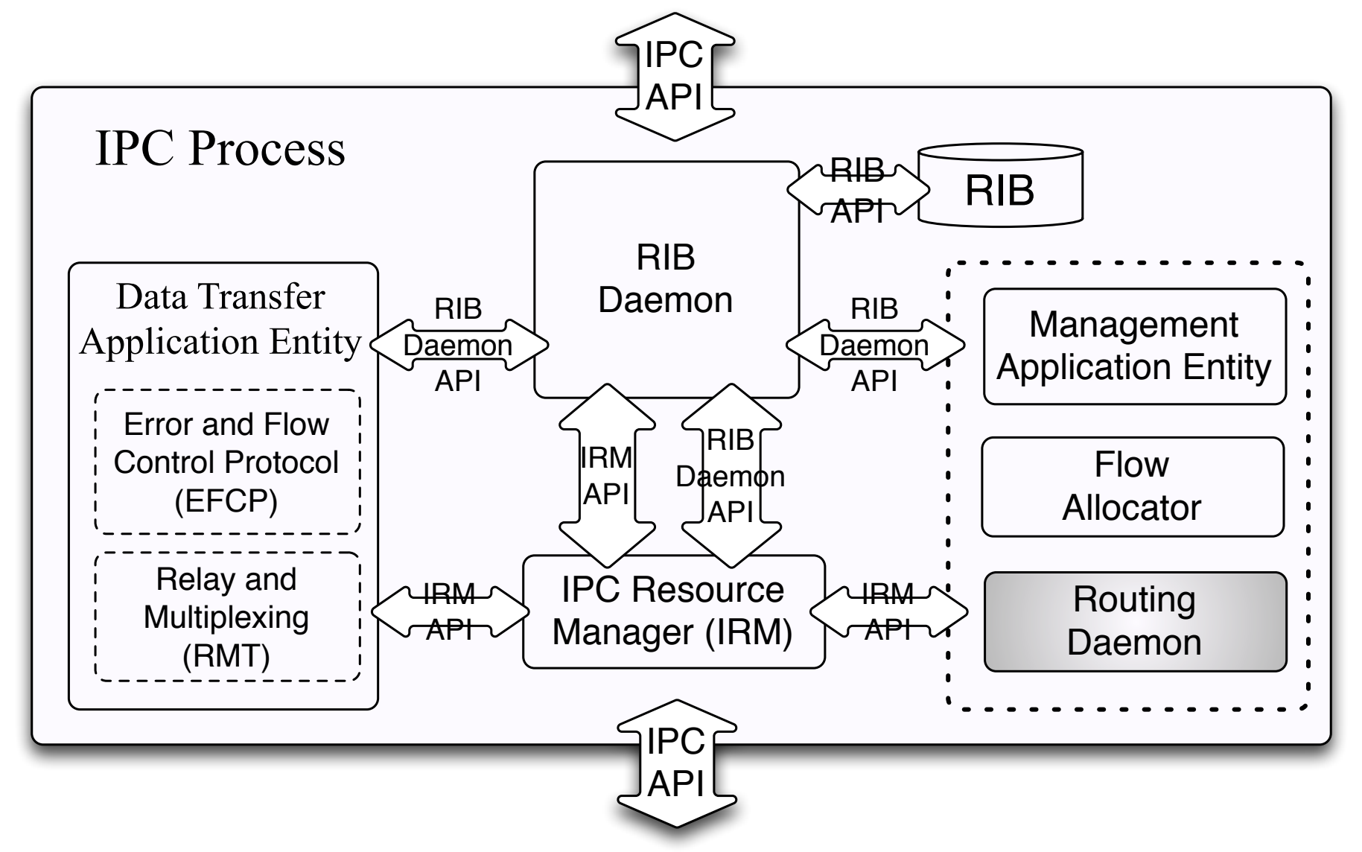
IPC Process
IPC API
RIB API
RIB
RIB Daemon
Data Transfer Application Entity
Error and Flow Control Protocol (EFCP)
Relay and Multiplexing (RMT)
RIB Daemon API
IRM API
IPC Resource Manager (IRM)
Management Application Entity
Flow Allocator
Routing Daemon

Fig 3 : IPC components and RINA APIs

Video Streaming on Legacy Hosts

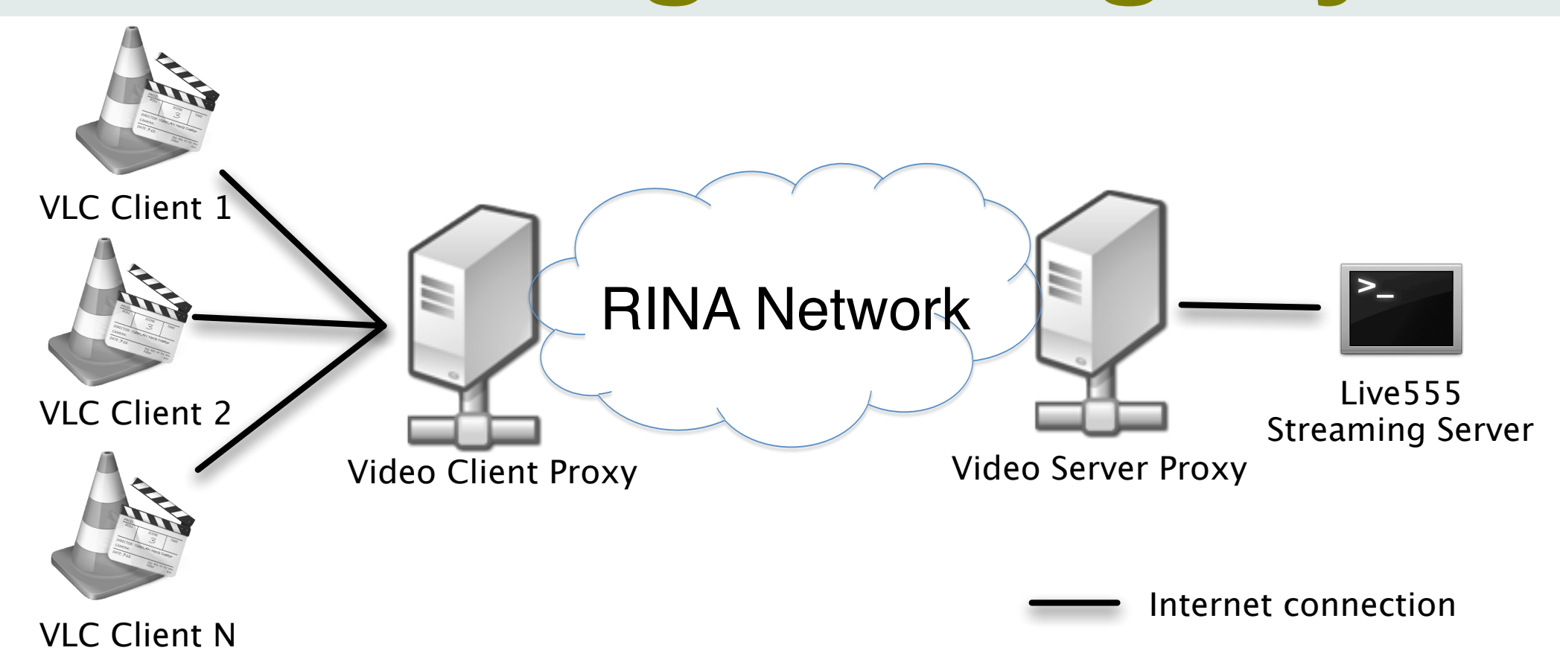

Fig 4 : Video clients (VLC players) are connected to the video streaming server through RINA proxies which are connected over a RINA network. The RINA network is composed of DIFs, where routing policies (and other policies) can be easily configured for each DIF

Resources Reserved from GENI

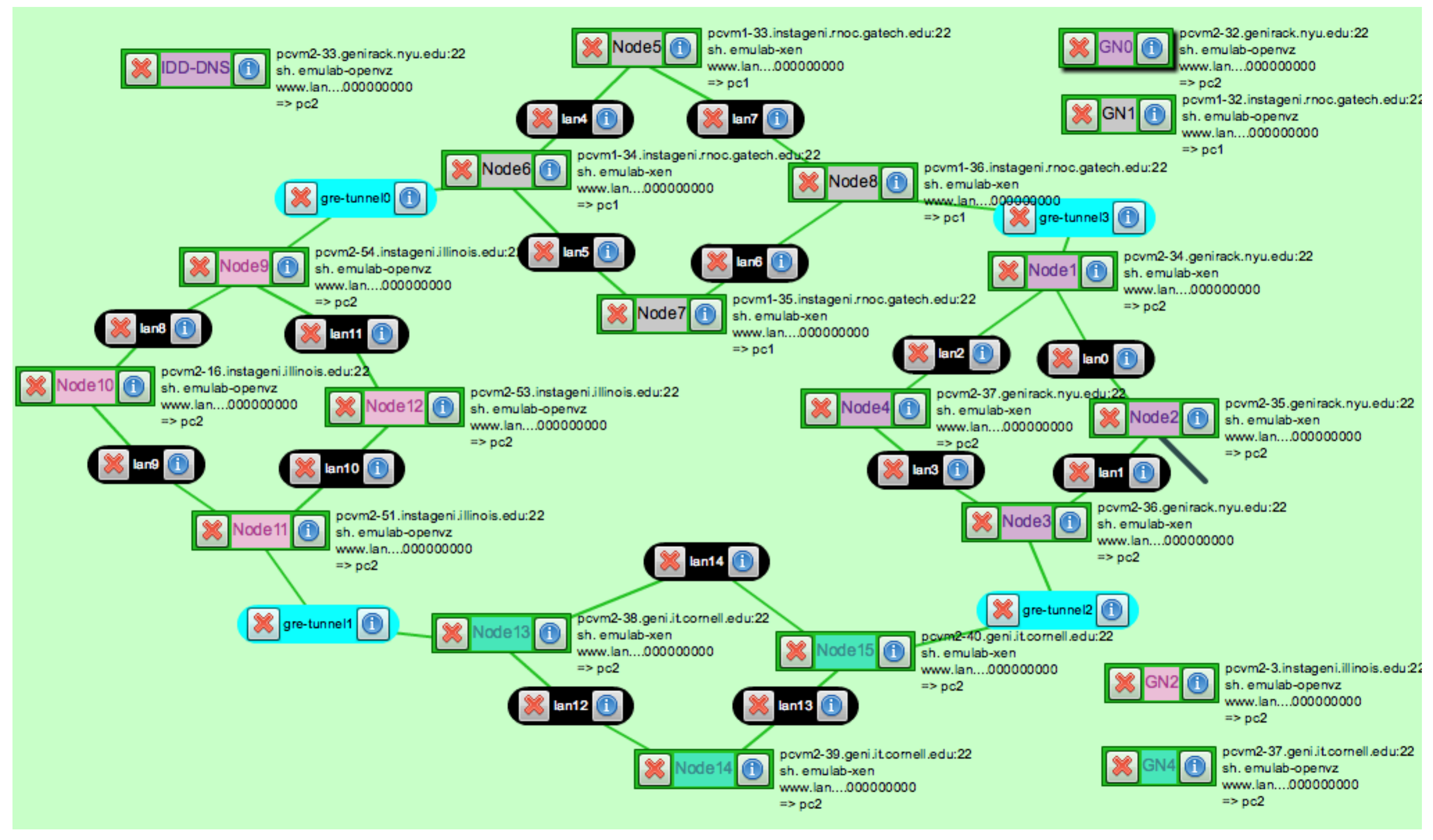

Fig 5 : VMs from four InstaGENI aggregates (Gatech, NYU, Cornell and UIUC), and VMs in different aggregates are connected using GRE tunnels. Each RINA node is running on a VM

Programming Networking Policies

- ProtoRINA enables policies to be configured easily
- We demonstrate two policies: DIF topology-formation policy and link-cost routing policy

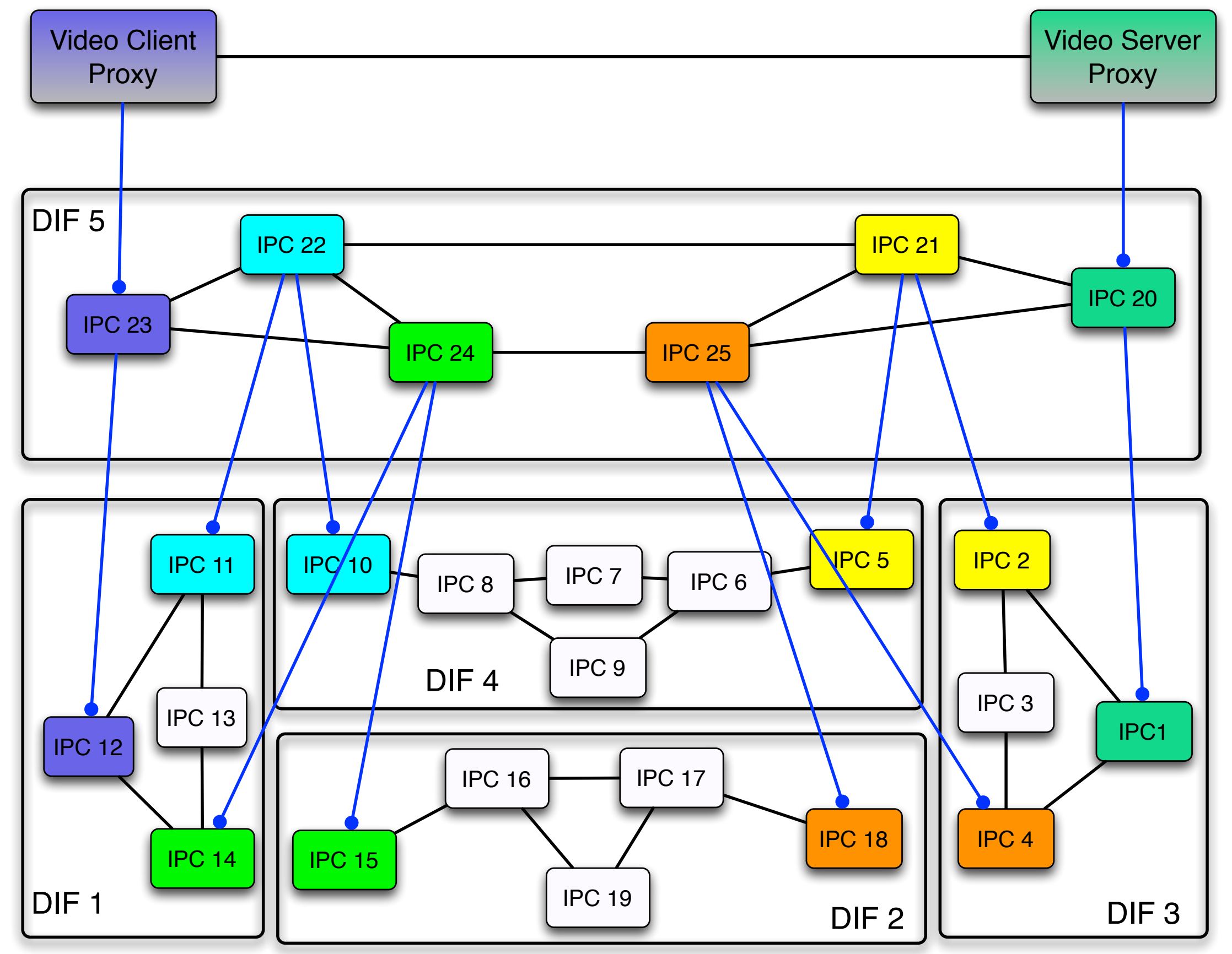

Fig 6: Two-level DIF topology

Experimental Results

- Goal: Reduce end-to-end jitter
- Form a two-level DIF topology where processes in the higher-level DIF reside across aggregates
- In the lower-level DIF, link-cost policy is hop
- In the higher-level DIF, change link-cost policy from hop to jitter

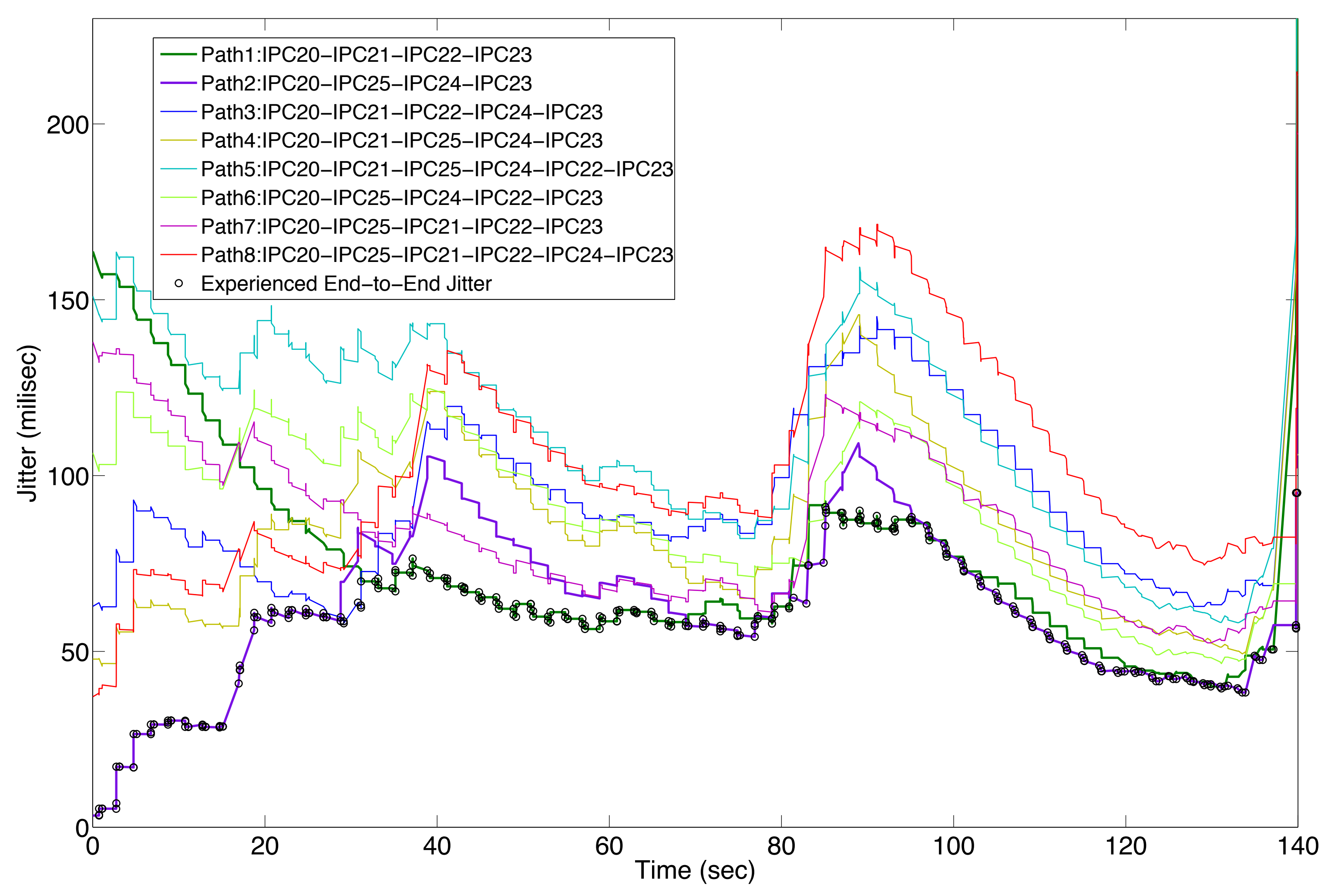

Fig 7: End-to-end jitter tracks the least path jitter

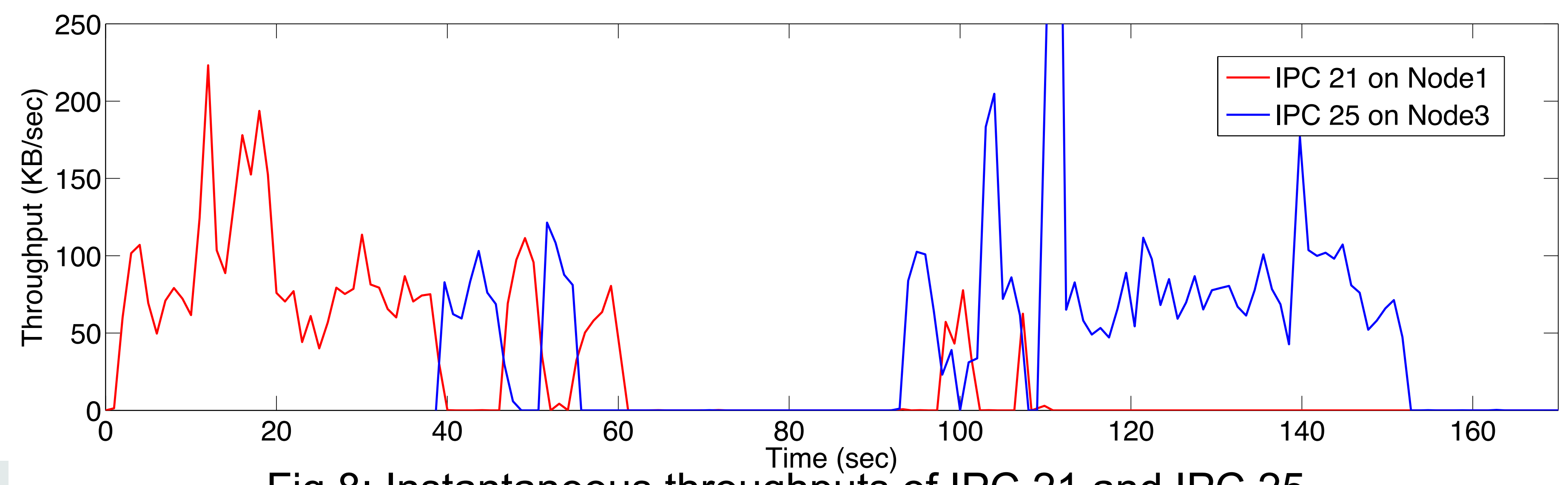

Fig 8: Instantaneous throughputs of IPC 21 and IPC 25

Experiences using GENI

- Reserving resources over multiple aggregates using GRE tunnels is more likely to succeed as compared with stitching, which is less reliable
- A large MTU can degrade TCP performance over GRE tunnels
- GENI Desktop facilitates real-time monitoring of experiments
- GENI user mailing list and GENI Desktop mailing list are really helpful

References

[1] John Day. "Patterns in Network Architecture: A Return to Fundamentals". Prentice Hall, 2008.

[2] John Day, Ibrahim Matta and Karim Mattar. "Networking is IPC: A Guiding Principle to a Better Internet". In ReArch 2008.

[3] Boston University RINA Lab. http://csr.bu.edu/rina.

[4] Yuefeng Wang, Ibrahim Matta, Flavio Esposito and John Day. "Introducing ProtoRINA: A Prototype for Programming Recursive-Networking Policies." In ACM SIGCOMM CCR, July, 2014.

[5] ProtoRINA. http://csr.bu.edu/rina/protorina.