Recursive InterNetwork Architecture
An Assessment of the IRATI Implementation

Jeroen van Leur Jeroen Klomp

University of Amsterdam
System and Network Engineering

February 1, 2016

Research goals

Research question

What is the current state of the IRATI RINA implementation?

- Find out which Recursive InterNetwork Architecture (RINA) implementations exist
- Find out their differences
- Find out how an experimental network needs to be set up
- Find out how resilient the routing in a small network is

Problems with TCP/IP

- Mobility not straightforward
- Multihoming does not scale
- Multicast does not scale
- Quality of Service does not scale
- Many security issues

What causes these problems?

- TCP/IP has an incomplete addressing scheme
  - Applications are not named
  - IP addresses name the interface, not the node
  - Point of attachment (link-layer) addresses are in concept the same as IP addresses
- No integrated security

What is wrong with the layers?

- Layers not properly defined and inflexible

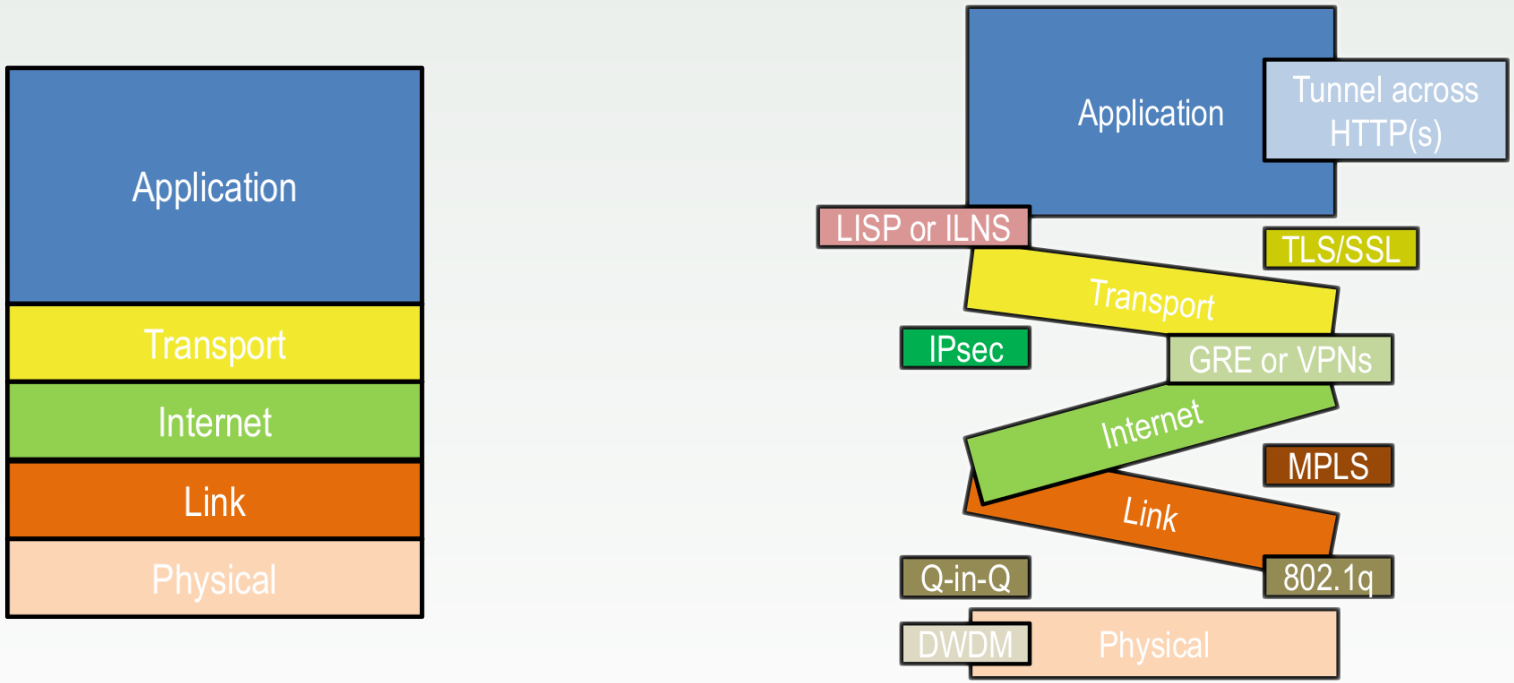


Figure: TCP/IP model?![1]

[1](Veselý, Marek, Hykel, & Ryšavý, 2015)

Layers in RINA

"The Internet is an unfinished demo" — John Day (2008)

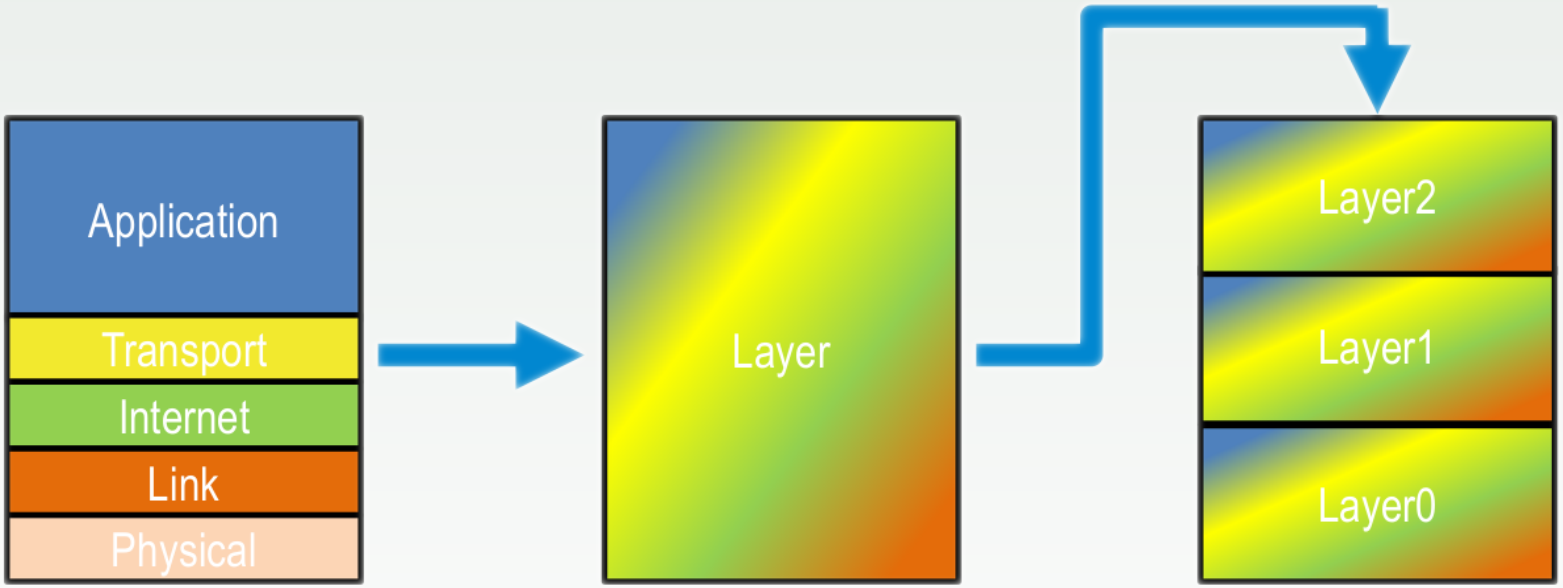


Figure: RINA's recursive layered approach[2]

[2](Veselý et al., 2015)

RINA concepts

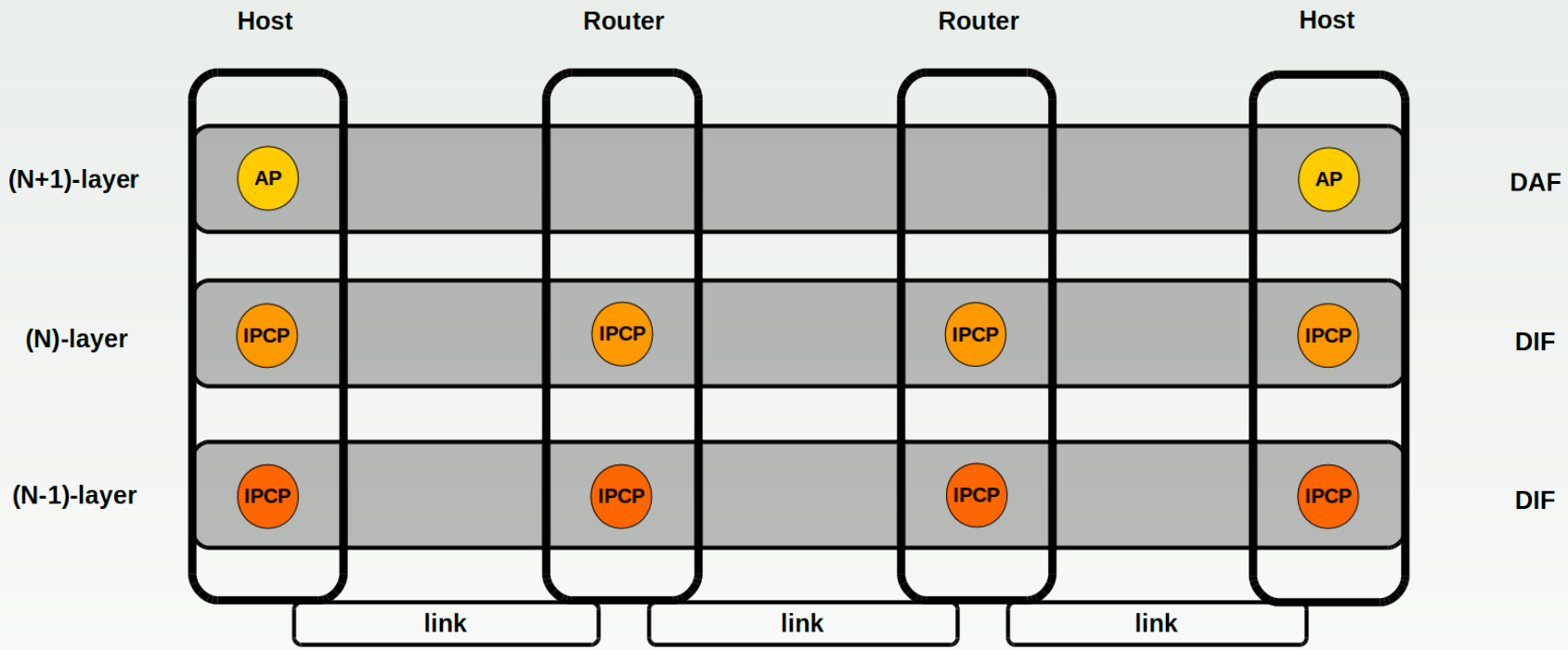


Figure: RINA layers and components[3]

[3] Based on (Grasa et al., 2011)

Communication in RINA

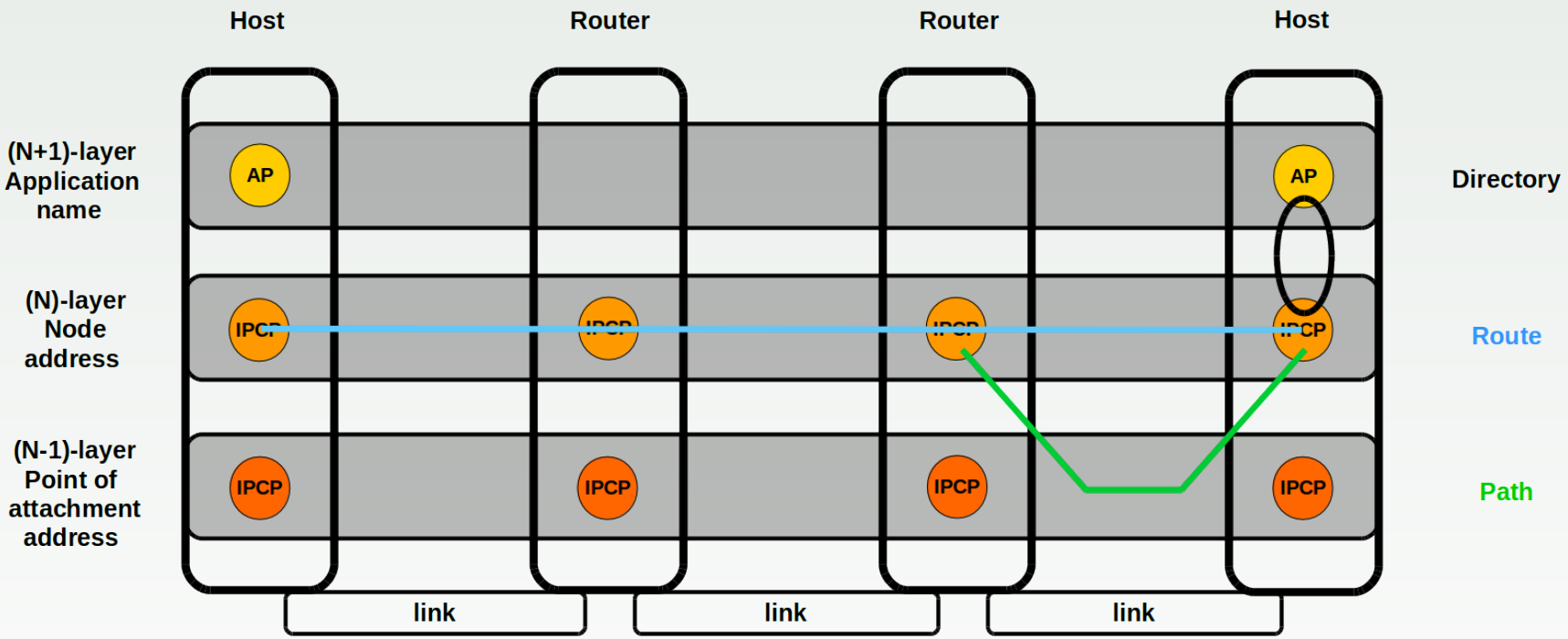


Figure: RINA directory, routes and paths[4]

[4] based on (Grasa et al., 2011)

RINA protocols

- Only two protocols
  - Error and Flow Control Protocol
    - Provides both unreliable (DTP)[5] and reliable (DTCP)[6] flows
    - No need for handshakes
    - Flows distinguished by Connection-ID
  - Common Distributed Application Protocol
    - Object-based communication
    - Only six primitive operations: Create/Delete, Read/Write, Start/Stop

[5] Data Transfer Protocol

[6] Data Transfer Control Protocol

Implementations

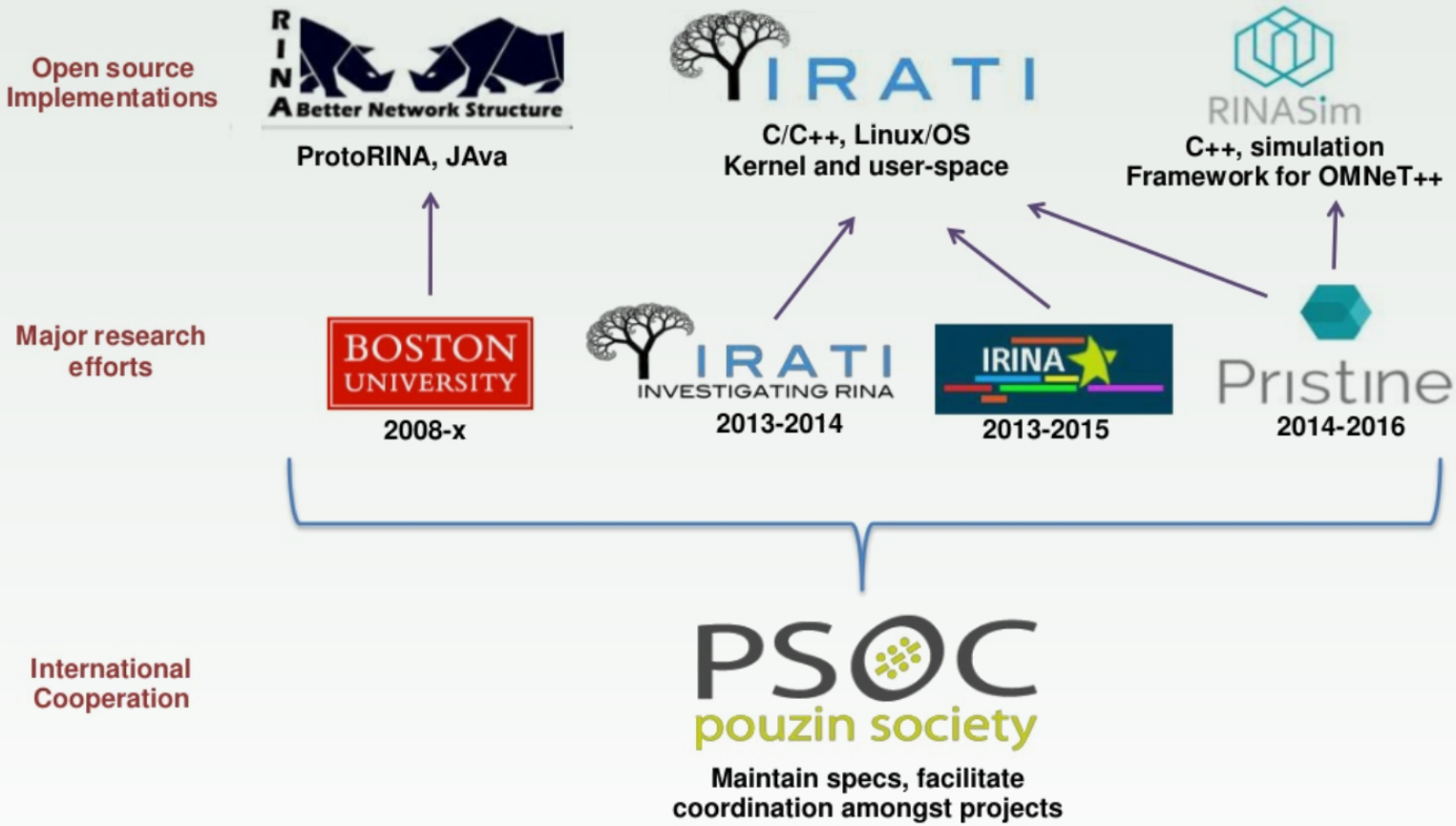


Figure: PSOC overviewed RINA implementations[7]

[7] Adapted from (Grasa, 2015)

IRATI

- Multiple shim Distributed IPC Facilities (DIFs)
  - UDP/IP
  - Ethernet via 802.1Q
  - Hypervisor to guest
  - Dummy shim for debugging
- Routing
  - Intermediate System-to-Intermediate System (IS-IS)
  - IP Fast Reroute (IPFRR)
  - Optional multipath routing with equal-cost multipath routing (ECMP) plugin

Design

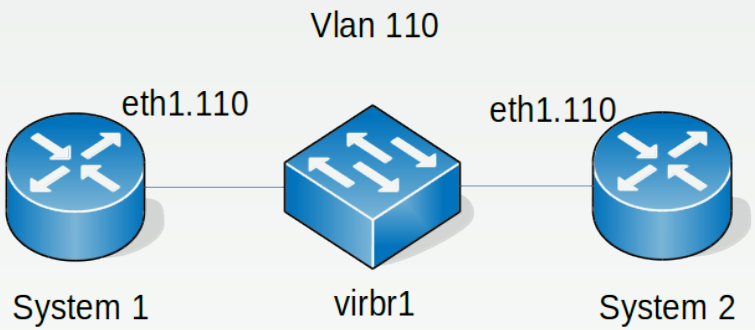


Figure: Physical network design

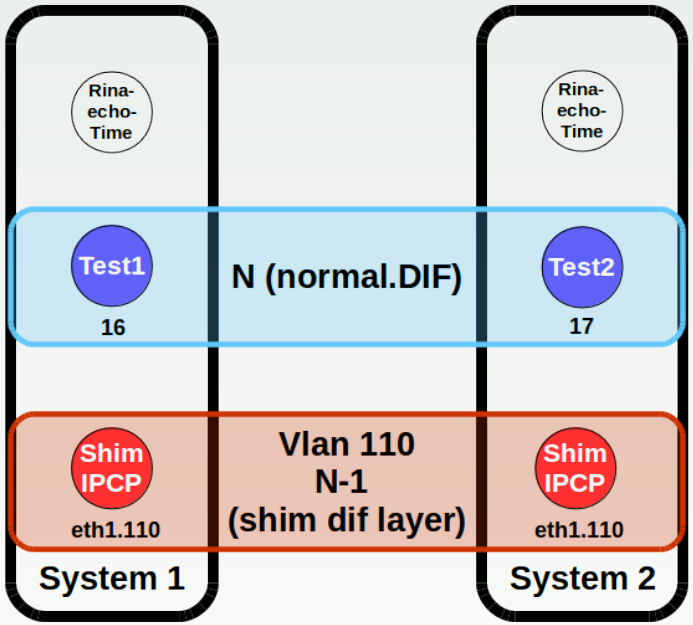


Figure: Logical network design

Basic tests

- IRATI stack
  - Initialisation
  - Enrolling to DIF
- Connectivity test
  - Behaviour of flow
  - Monitoring the connectivity
- Performance test

Results

- Susceptible to configuration errors
- Debugging options: high I/O and impact CPU
- Tooling results:
  - Echo tool shows response round-trip time (RTT) less than 1 ms.
  - Wireshark showed src/dst address correctly after patching
  - Performance tests results from 400 Mbit/sec to 15 Gbit/sec

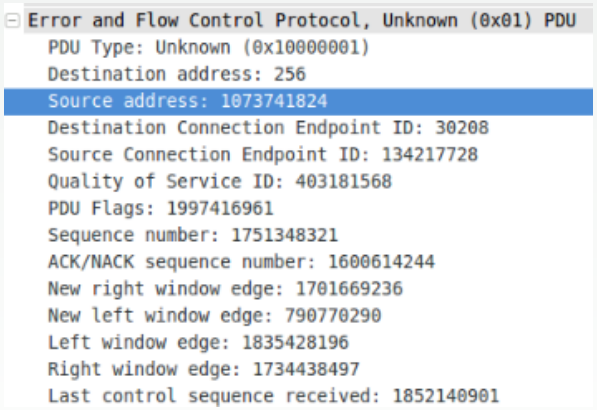


Figure: Wrong address

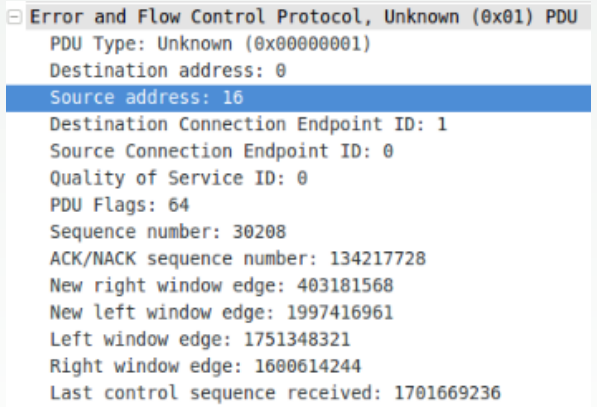
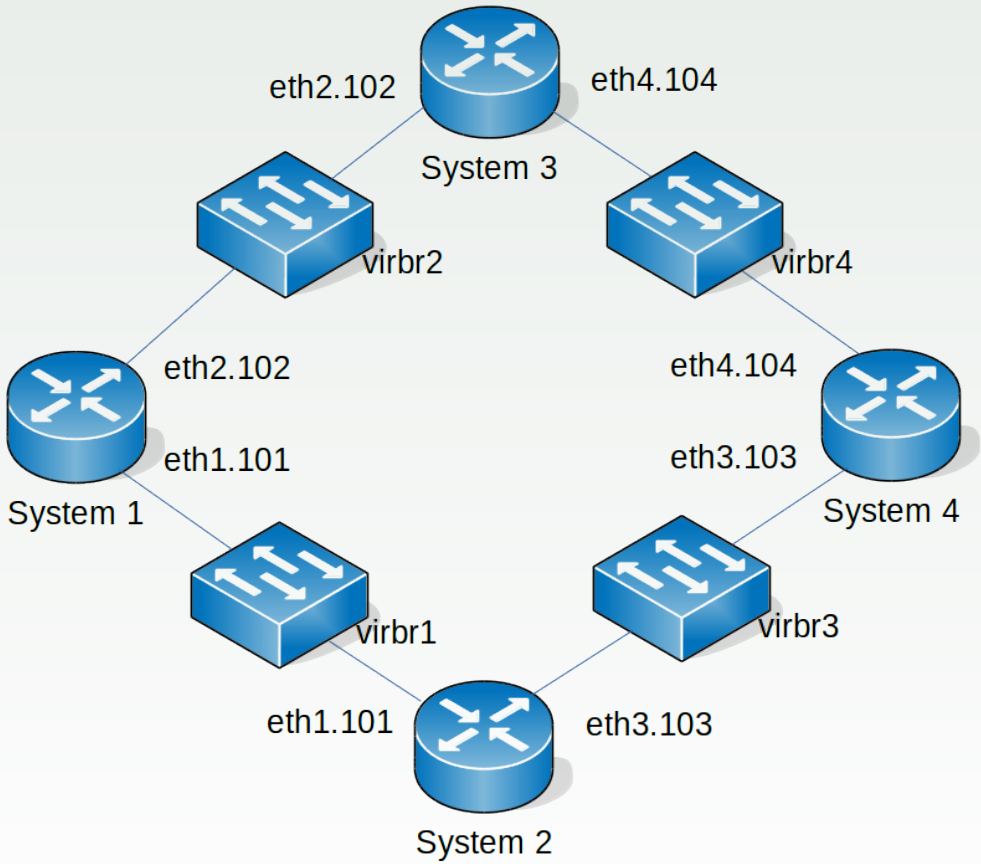
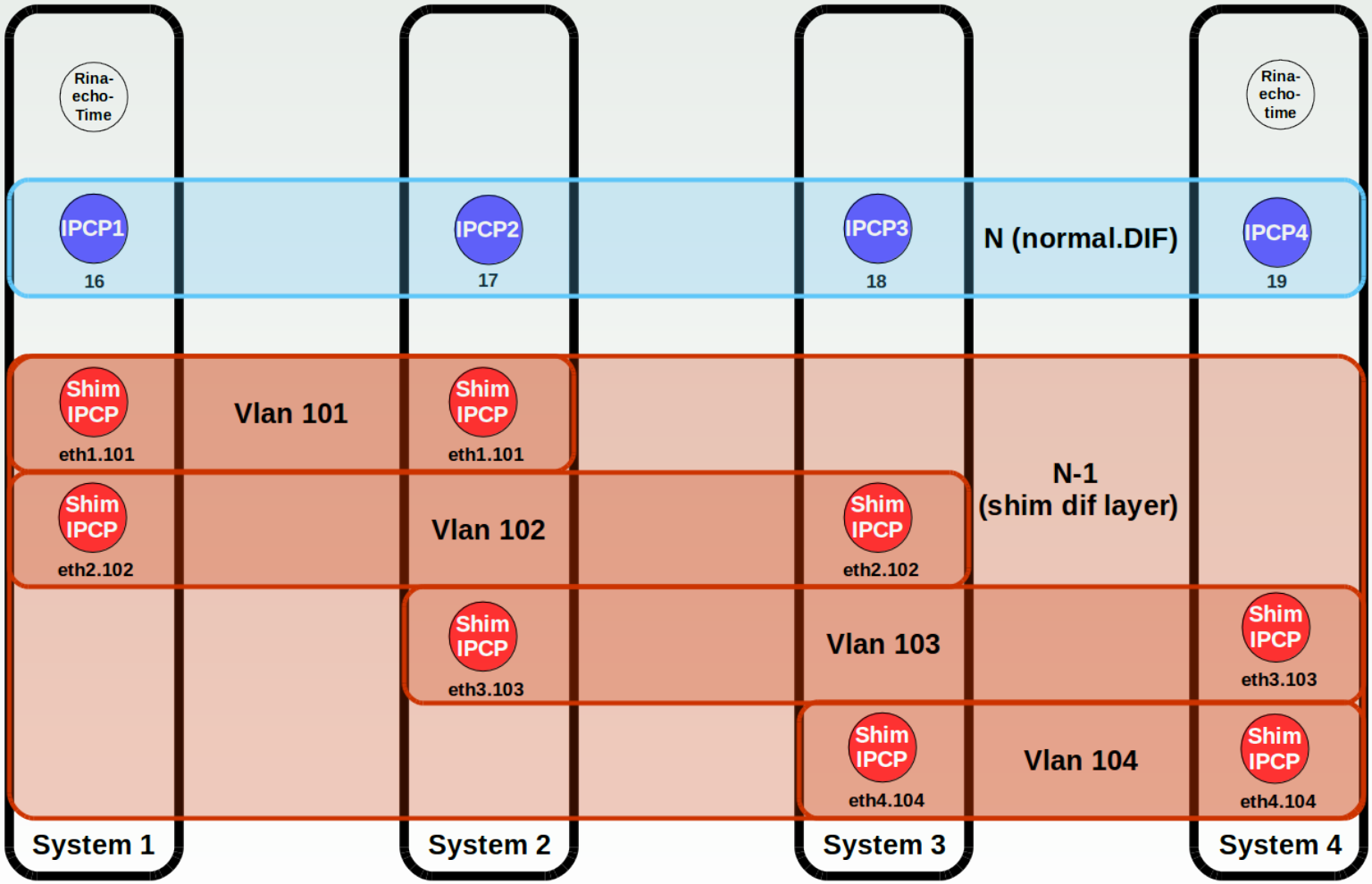


Figure: Correct address

Physical design



Logical design



Routing tests

- Configuration
  - Enrolling to the DIFs
  - Changes in the tools
- Resilience tests
  - Disconnecting links
  - Connectivity test
- Multipath plugin
- Performance test

Results

- Manual configuration of all systems
- Routing information in resource information base (RIB)
  - Next Hops
  - Underlying DIF
  - All neighbours
- Network updates are propagated

Next hops

```
Name: /resalloc/nhopt/key=16-0; Class: NextHopTableEntry; Instance: 47
Value: Destination address: 16; QoS-id: 0; Cost: 1; Next hop addresses: 17 /

Name: /resalloc/nhopt/key=17-0; Class: NextHopTableEntry; Instance: 48
Value: Destination address: 17; QoS-id: 0; Cost: 1; Next hop addresses: 17 /

Name: /resalloc/nhopt/key=18-0; Class: NextHopTableEntry; Instance: 49
Value: Destination address: 18; QoS-id: 0; Cost: 1; Next hop addresses: 18 /
```

Routing Resiliency

- System 1 - System 2 disconnected
- No re-routing possible for existing and new flows
- Multipath plugin
  - Multiple paths in Wireshark
  - Next hops change in RIB
  - Lacks link failure resiliency

Multipath next hops

```
Name: /resalloc/nhopt/key=16-0; Class: NextHopTableEntry; Instance: 47
Value: Destination address: 16; QoS-id: 0; Cost: 1; Next hop addresses: 17/

Name: /resalloc/nhopt/key=17-0; Class: NextHopTableEntry; Instance: 48
Value: Destination address: 17; QoS-id: 0; Cost: 1; Next hop addresses: 18/

Name: /resalloc/nhopt/key=18-0; Class: NextHopTableEntry; Instance: 49
Value: Destination address: 18; QoS-id: 0; Cost: 1; Next hop addresses: 17/ 18
```

Conclusion

- IRATI is still in an experimental phase
- Routing was not resilient
- Using IRATI requires Unix background and programming skills to debug issues
- Ongoing progress:
  - Future projects will enhance IRATI
  - New ProtoRINA release this year
  - Active improvement of the RINA reference model

Any questions?

References

Grasa, E. (2015, October). Rina essentials. NEXTWORKS, PRISTINE, and University of Pisa. Retrieved January 27, 2016, from http://ict-pristine.eu/wp-content/uploads/2015/11/IF2015-SDN-NFV-RINA-04_RINA-essentials.pdf

Grasa, E., Trouva, E., Phelan, P., de Leon, M. P., Day, J., Matta, I., ... Bunch, S. (2011). Design principles of the recursive internetwork architecture (RINA). Retrieved January 29, 2016, from http://www.future-internet.eu/fileadmin/documents/fiarch23may2011/06-Grasa_DesignPrinciplesOTheRecursiveInterNetworkArchitecture.pdf

References

Veselý, V., Marek, M., Hykel, T., & Ryšavý, O. (2015). Rinasim: your recursive internetwork architecture simulator. September 3, 2015 (7). Omnet++ community summit 2015. Zurich. Retrieved January 6, 2016, from https://summit.omnetpp.org/archive/2015/#keynotes