RINASim: YOUR RECURSIVE INTERNETWORK ARCHITECTURE SIMULATOR

Vladimír VESELY, Marcel MAREK, Tomáš HYKEL, Ondřej RYŠAVÝ

2ND OMNeT++ SUMMIT
3RD-4TH SEPTEMBER 2015, ZÜRICH, SWITZERLAND
**Theory**

- *Your teachers told you about this…*
  - 1978 – 1994

**ISO-OSI RM**

- Application
- Presentation
- Session
- Transport
- Network
- Data-link
- Physical
...or this...

1983

TCP/IP RM

Application

Transport

Internet

Link
Theory

...or that!

TCP/IP RM

- **Application**: network part of each application
- **Transport**: data transfer services
- **Internet**: logical communication
- **Link**: adjacent communication
- **Physical**: medium abstraction
Have you noticed clash of theory and practice?

TCP/IP RM

Application
Transport
Internet
Link
Physical
DWDM
THEORY

Have you noticed clash of theory and practice?

TCP/IP RM

Application

Transport

Internet

Link

DWDM

Physical

802.1q
Have you noticed clash of theory and practice?

**Theory**

TCP/IP RM

- Application
- Transport
- Internet
- Link
- Q-in-Q
- 802.1q
- DWDM
- Physical
Have you noticed clash of theory and practice?

TCP/IP RM

Application
Transport
Internet
Link
IPsec

802.1q
Q-in-Q

Physical
DWDM
Have you noticed clash of theory and practice?
**THEORY**

Have you noticed clash of theory and practice?

TCP/IP RM

- Application
- Transport
- Internet
- Link
- Physical
- Q-in-Q
- 802.1q
- MPLS
- IPsec
- TLS/SSL
Have you noticed clash of theory and practice?

TCP/IP RM

Application

Transport

IPsec

Internet

GRE or VPNs

Link

MPLS

Q-in-Q

802.1q

DWDM

Physical
Theory

Have you noticed clash of theory and practice?

TCP/IP RM

Application

- LISP or ILNS
- TLS/SSL
- Transport
- IPsec
- GRE or VPNs
- Internet
- MPLS
- Link
- Q-in-Q
- 802.1q
- DWDM
- Physical
Have you noticed clash of theory and practice?
Open problems of nowadays Internet

- Multi-homing
- Identity
- Mobility
- Default-free Zone growth

What is unique address???
RECURSIVE INTERNETWORK ARCHITECTURE

- One generic layer (called DIF, Distributed IPC Facility) that limits scope

1) Data transfer with soft-state timer-based synchronization
2) Complete addressing and naming scheme
3) Split between mechanism (fixed) and policy (flexible)
4) Single generic application protocol
History

- In 2007 book
  Patterns in Network Architecture: A Return to Fundamentals
  by John Day
1) **Soft-State Transport Protocol**

- Hard-state = explicit synchronization prior to communication using special messages (i.e., TCP flags and 3-way handshake)

- **Data Transfer Protocol (DTP)** based on Richard Wattson’s transport protocol **Delta-t** from 1984
  - Proves that hard-state is unnecessary IFF
    - **Maximum Packet Lifetime** denotes upper bound time (value $MPL$) that a packet can exist in a network
    - **Retransmission-timer** specifies maximum period (value $R$) that a sender is willing to retransmit its unacknowledged messages;
    - **Acknowledgment-timer** defines maximum delay (value $A$) that the receiver of data can wait before sending acknowledgment.
    
    \[
    \Delta t = MPL + R + A \]

- All connections exist all the time
  - Decoupling of port allocation from synchronization
  - After period of 2-3 $\Delta t$ of no traffic, all synchronization state should be discarded, which effectively resets the connection

- Both reliable and unreliable transfer based on employment of sequence numbers
2) **COMPLETE ADDRESSING**

- Logical vs. Physical addresses
- Variable addresses
- Reflects scope (size) and usage (flat vs. hierarchical)
3) **MECHANISM vs. POLICY**

- i.e., error-checking (fixed) employing CRC (flexible)
- Programmable behavior using well-defined APIs
4) CDAP

- **Common Distributed Application Protocol (CDAP)**
  - *All application protocols are stateless, the state is in the application*
  - DTP modify state **internal** to the protocol and CDAP modify state **external** to the protocol
    - CDAP encapsulated into DTP
  - Request-response scheme with feedback

- **Primitive operations governing objects (i.e., files)**
  - Read / Write
  - Create / Delete
  - Start / Stop
    - 6 messages × 2 directions = 12 message types

- **Subcomponents**
  - CACE – connection establishment
  - Auth – authentication
  - CDAppP – protocol operation
CONCLUSION

- RINASim is full-fledged RINA simulating framework
  - Independent on other libraries (such as INET)
  - 105 899 SLOC and getting bigger
    - including *.h, *.cc, *.msg, *.ned, *.ini, *.xml
    - excluding comments and empty lines
- Complete recursive stack implementation
  - Generic CDAP socket-like APIs
  - Data transfer protocol
  - Routing protocol policies
  - Flow lifecycle
Thank you! Feel free to ask any question…

- **https://github.com/kvetak/RINA**
  - Expecting first research papers to be accepted
  - Work more on Enrollment, EFCP
  - Develop a new routing protocols with load-balancing features
  - 4th International RINA Workshop in April 2016 hosted in Brno

- **https://github.com/kvetak/ANSA**
  - Submit a new INET3.0 contributions
    - EIGRP
    - LISP
    - VRRP
    - TRILL and IS-IS