

# R2P2: Making RPCs first-class datacenter citizens

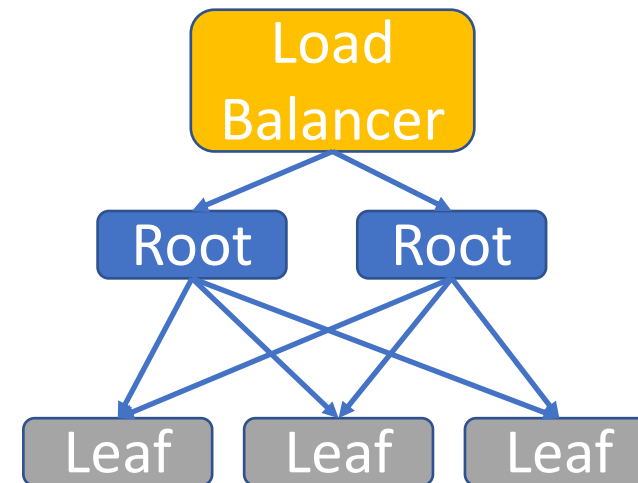
Marios Kogias George Prekas Adrien Ghosn Jonas Fietz  
Edouard Bugnion

The logo for EPFL (École Polytechnique Fédérale de Lausanne) is displayed in a bold, red, sans-serif font. The letters are stylized with a blocky, geometric appearance.

# Datacenter Communication



- Data-stores, search, etc...
- Complex Fan-in/Fan-out patterns
- $\mu$ s-scale Remote Procedure Calls
- Tight tail-latency SLOs



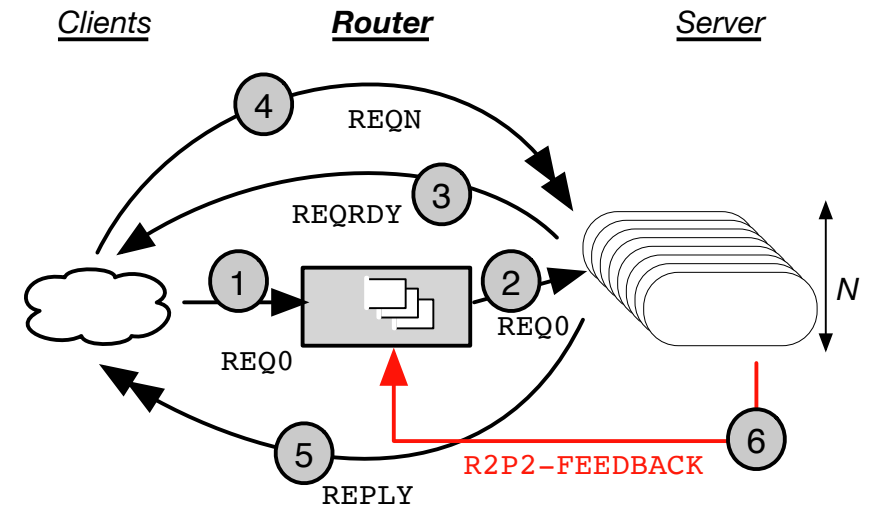
# RPC Multiplexing

- Use TCP (byte-stream-oriented) to send RPCs (message-oriented)
  - gRPC, Thrift, etc...
- End-host problems:
  - Head-of-line blocking
  - Scheduling
  - Buffer-bloat
  - Application-level RPC parsing
- Infrastructure-wide problems:
  - Load-balancing
    - Software-based L7 loadbalancing
  - Inability to leverage network programmability

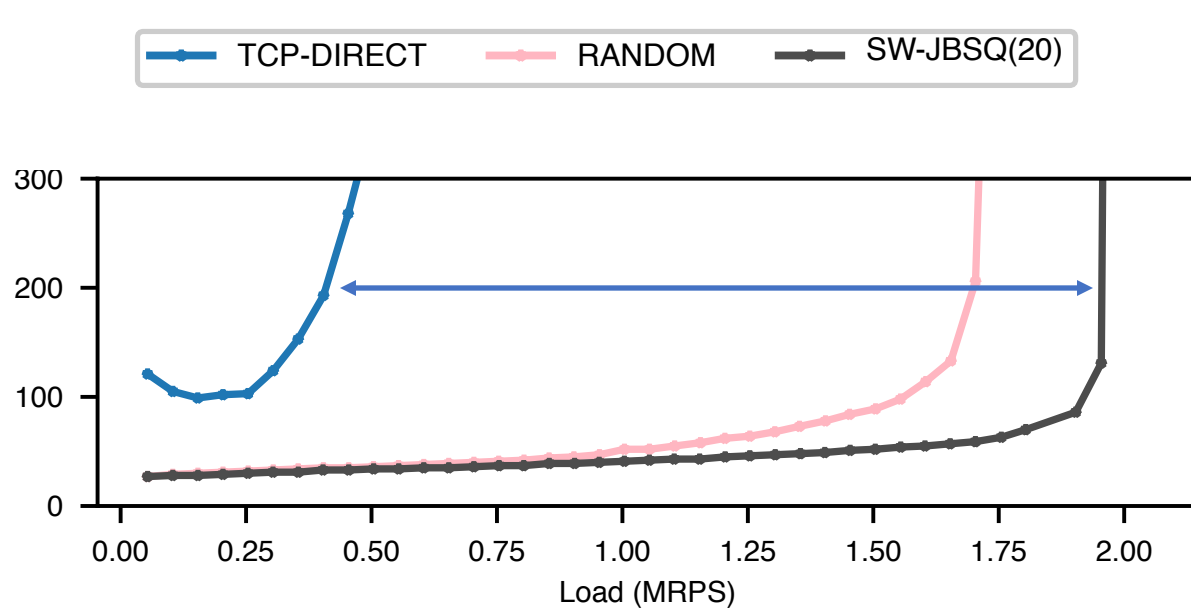
Let's make RPCs first class datacenter citizens

# R2P2 – Request Response Pair Protocol

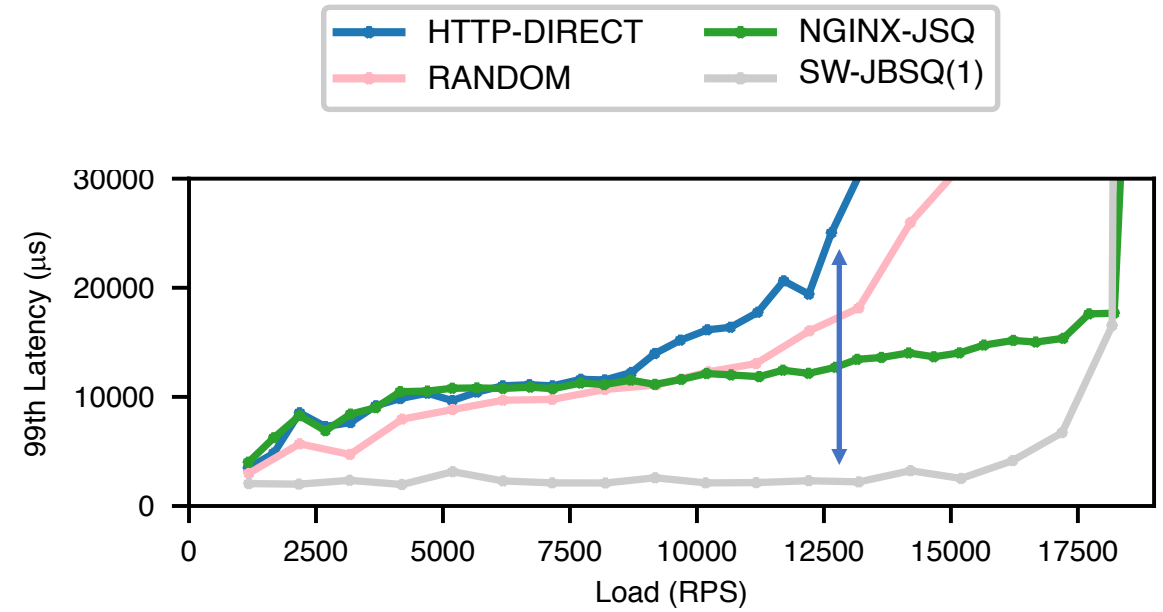
- Expose the RPC abstraction to the network and the end-hosts
- Independent request-response pairs
- Request/Response - aware middleboxes
- Decouple RPC target selection from request and reply streaming
- Use programmable switches for in-network RPC scheduling
  - JBSQ policy



# Key Results



Redis



Lucene

# R2P2: Making RPCs first-class datacenter citizens

Marios Kogias George Prekas Adrien Ghosn Jonas Fietz  
Edouard Bugnion

The logo for EPFL (École Polytechnique Fédérale de Lausanne) is displayed in a bold, red, sans-serif font. The letters are stylized with a blocky, geometric appearance.