STN: A Robust and Distributed Control Plane

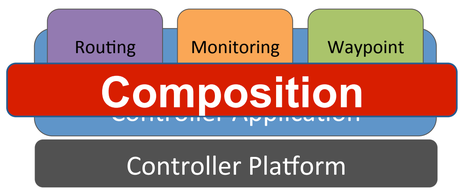
Marco Canini, Daniele De Cicco, Petr Kuznetsov, Dan Levin, Stefan Schmid and Stefano Vissicchio

Telekom Innovation Laboratories

- How to realize distributed **policy composition,** with**:**
- Support for **multi-authorship** and **transactional semantics**, that is:
- Robust to a number of controller **stop-failures**

SDN Policy Composition Review

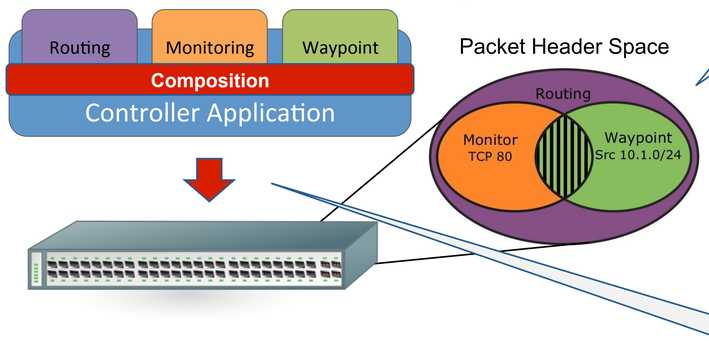
Policy may originate from **multiple authors**, defined across **multiple functional modules**.



Foster '11, Monsanto '13: Modular, parallel and sequential composition

...necessitates policy **composition prior to network update.**

How it all fits together

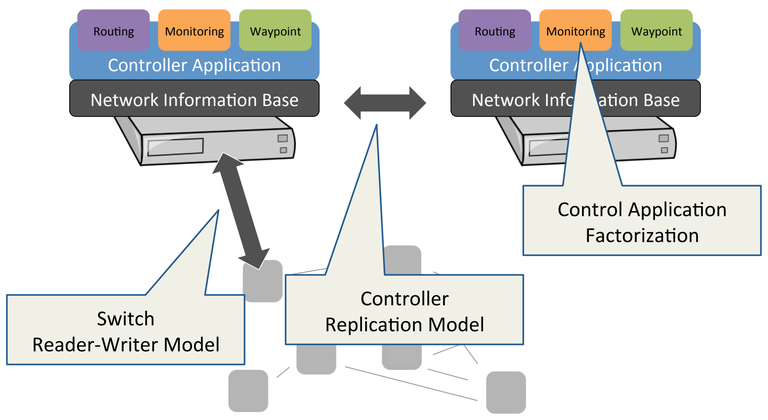


Ferguson '12,'13: Policy trees for multi-authorship

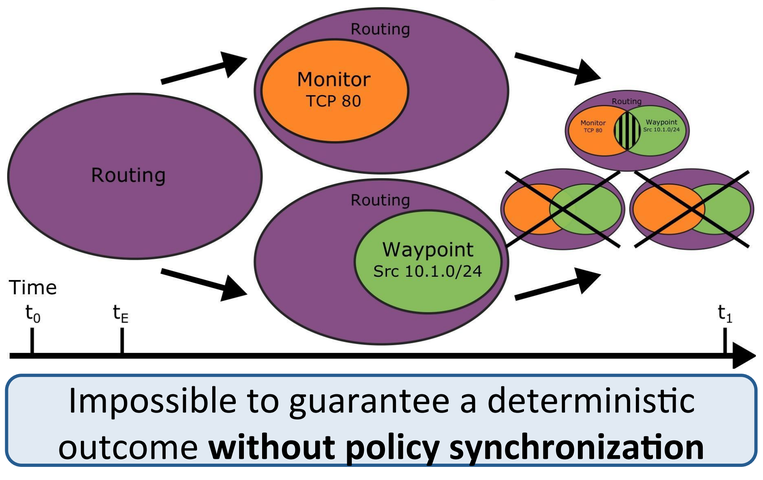
Reitblatt '12: Consistent network updates

1. Precedence must be defined across policy sources
2. Packet forwarding rule priorities must be defined, and respect policy source precedence

Problem: Distributed Policy Composition

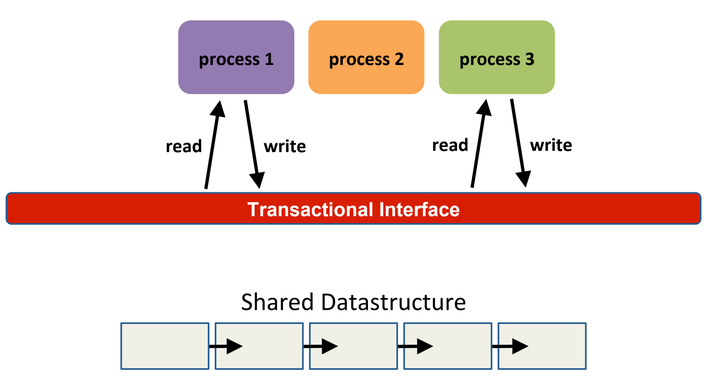


Concurrent Policy Composition Gone Wrong



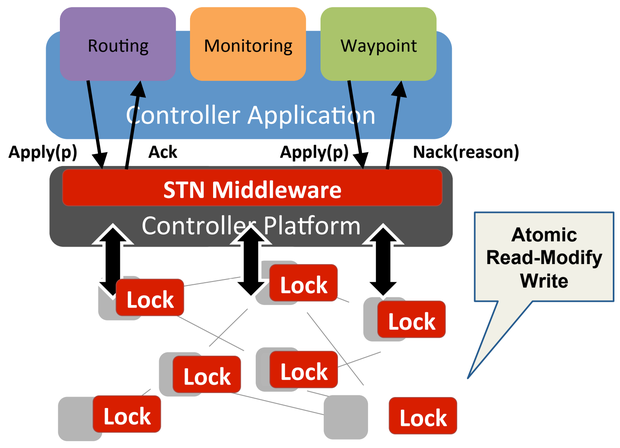
Impossible to guarantee a deterministic outcome **without policy synchronization**

Inspiration from Software Transactional Memory



STN: Software Transactional Networking

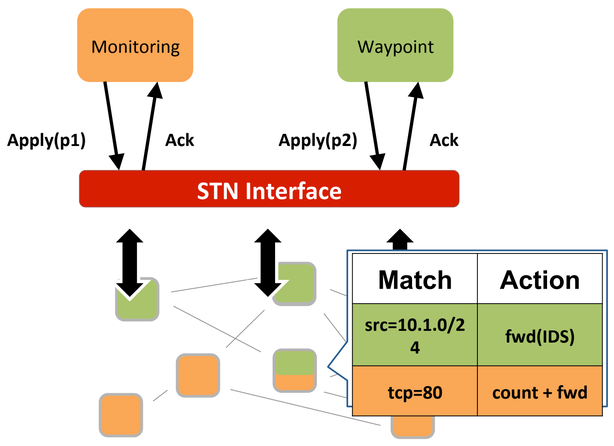
Conceptualizing STN



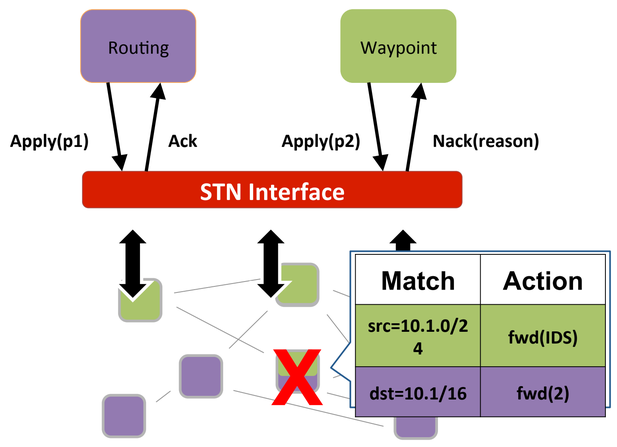
Prototype **implemented on pyretic** as an interface that provides:

- distributed **policy composition**
- support for **multi-authorship**
- **transactional all-or-nothing** policy composition semantics
- **per-packet consistent** policy updates

STN in Action (Ack Case)

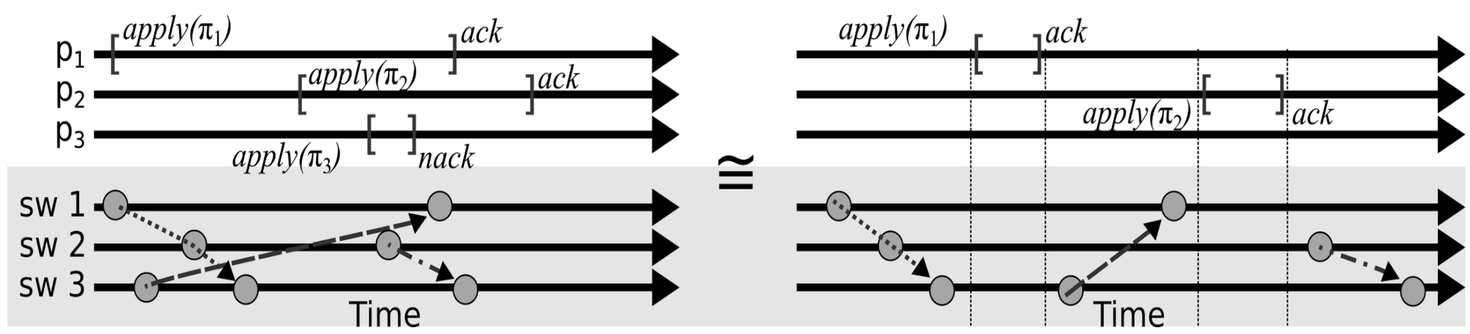


STN in Action (Nack Case)



STN prevents concurrent, conflicting policy updates from affecting any traffic

The Result: Linearizable Concurrent Policy Updates



Linearizability (an equivalent sequential history) is the "holy grail" safety property

Robustness to Controller "Stop-Failures"

Theorem 1

STN ensures linearizability and wait-freedom with exponential *tag complexity*

Theorem 2

STN is resilliant to f controller stop-failures with optimal tag complexity f+2

Wait-freedom is the "holy grail" liveness property

References

[1] Software Transactional Networking: Concurrent and Consistent Policy Composition, In Proceedings of SIGCOMM HotSDN 2013

[2] The Case for Reliable Software Transactional Networking, Research Report CoRR, http://arxiv.org/abs/1004.4701