

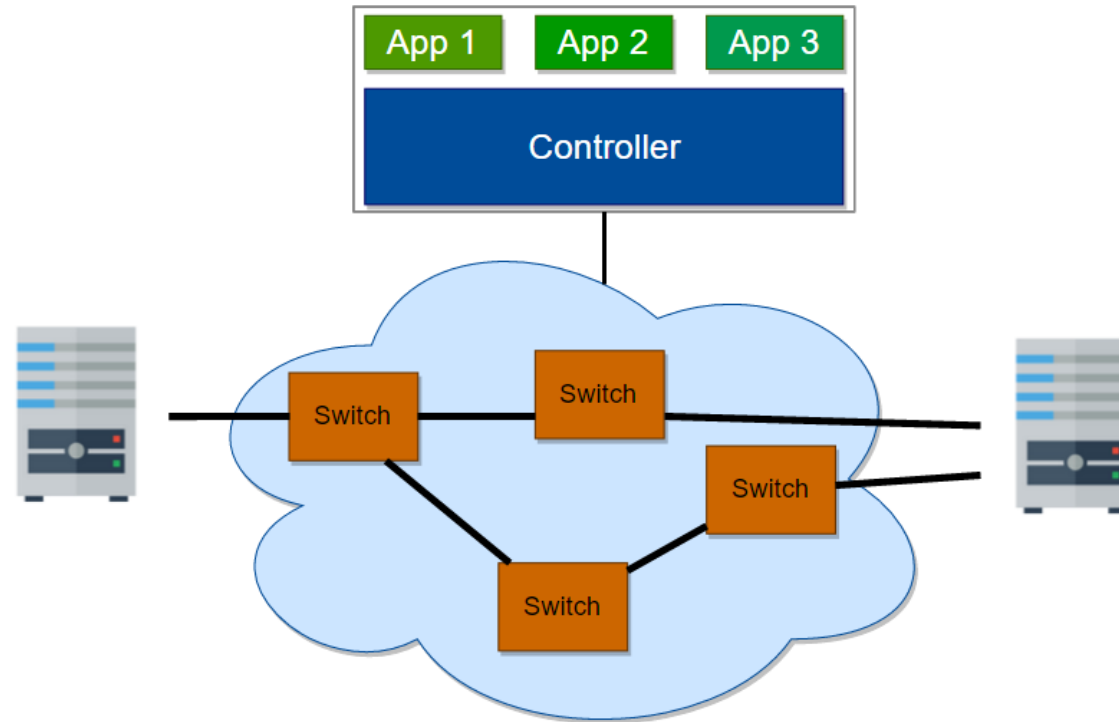


Distribution, fault-tolerance
& consistency in SDN control:
Pick three.

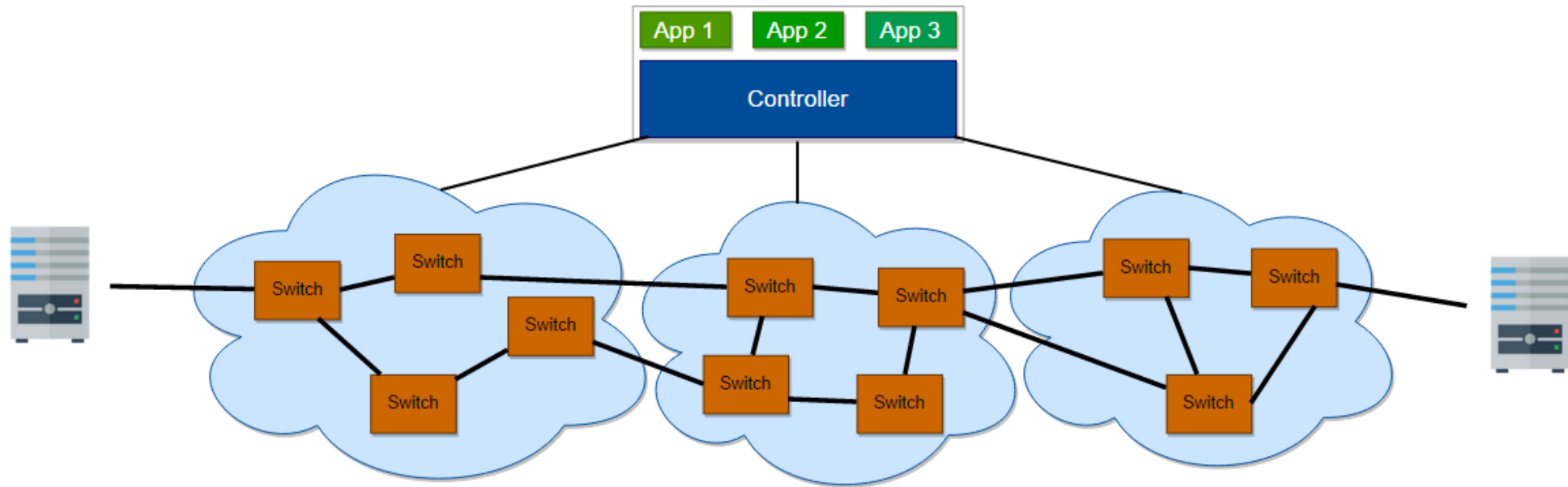
André Mantas



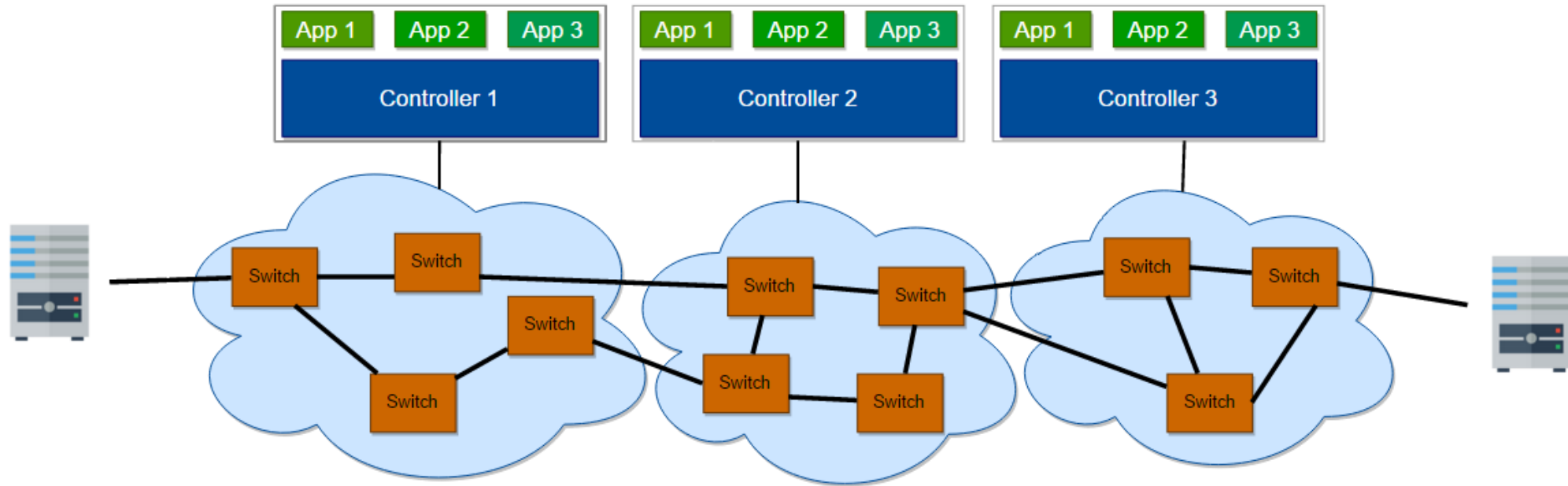
Context



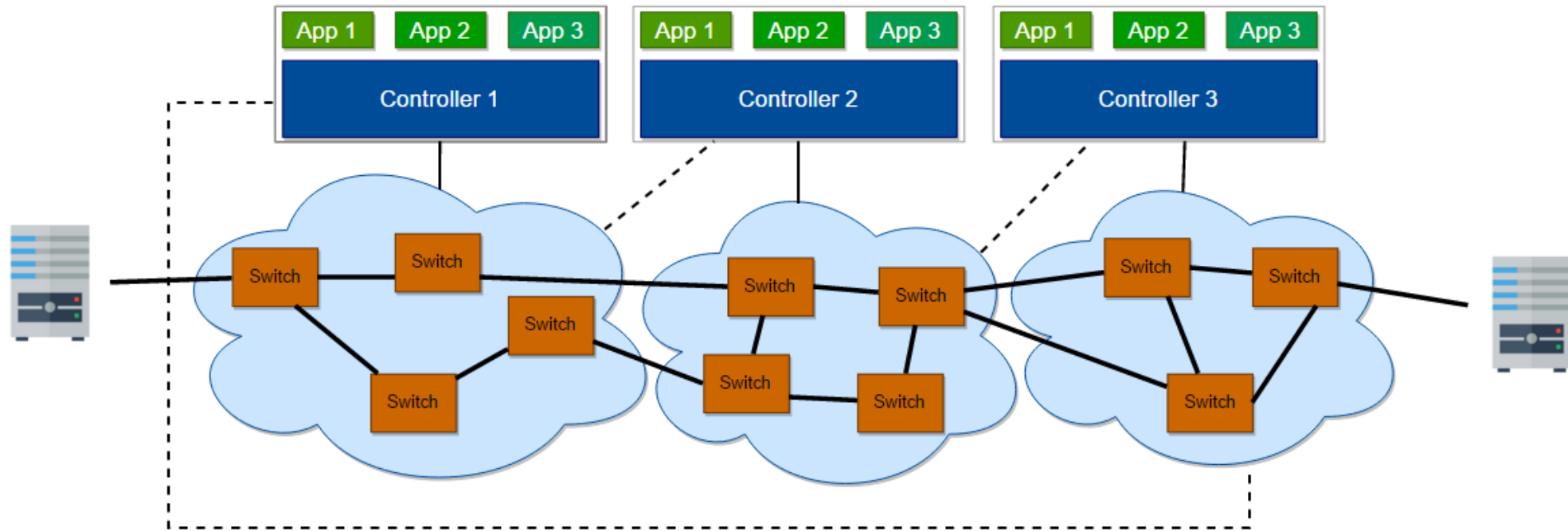
Context



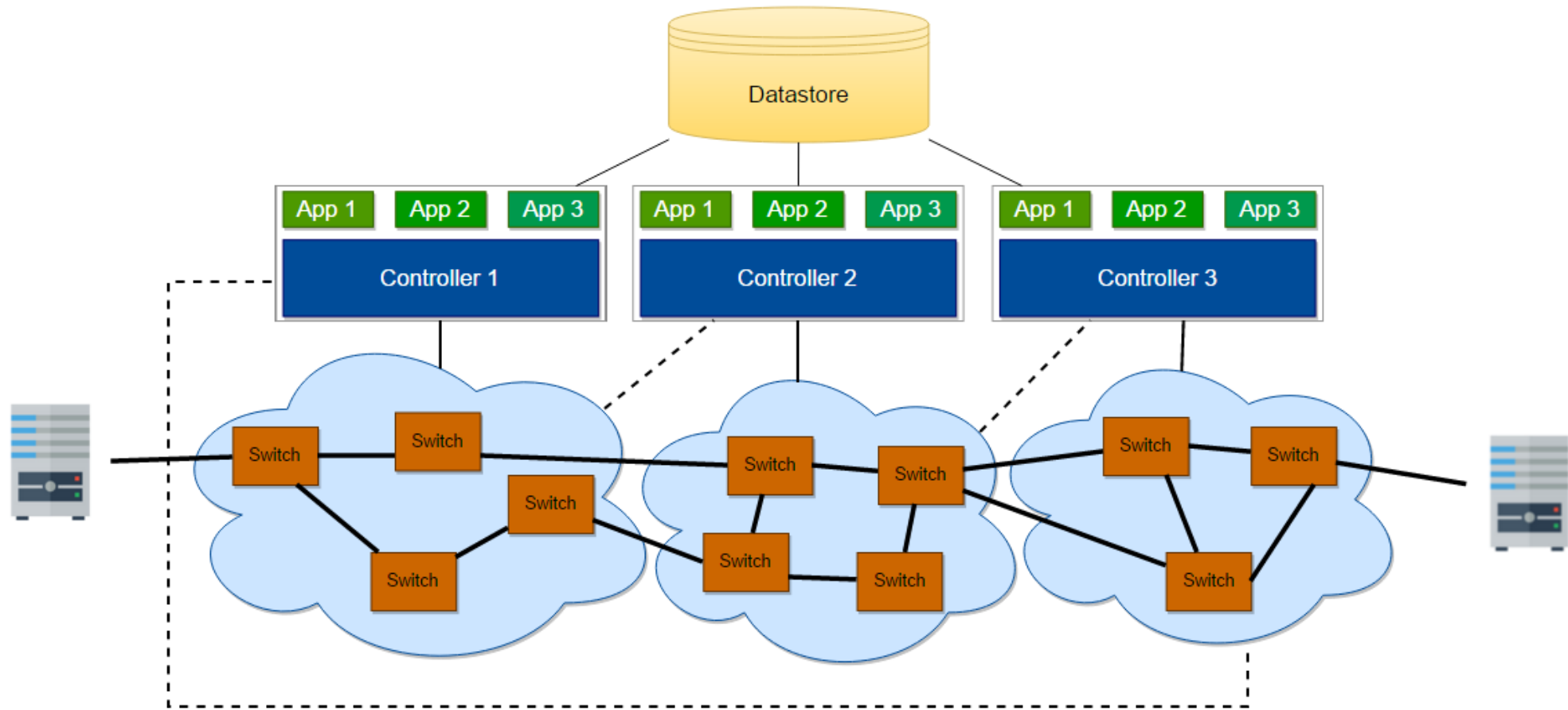
Context



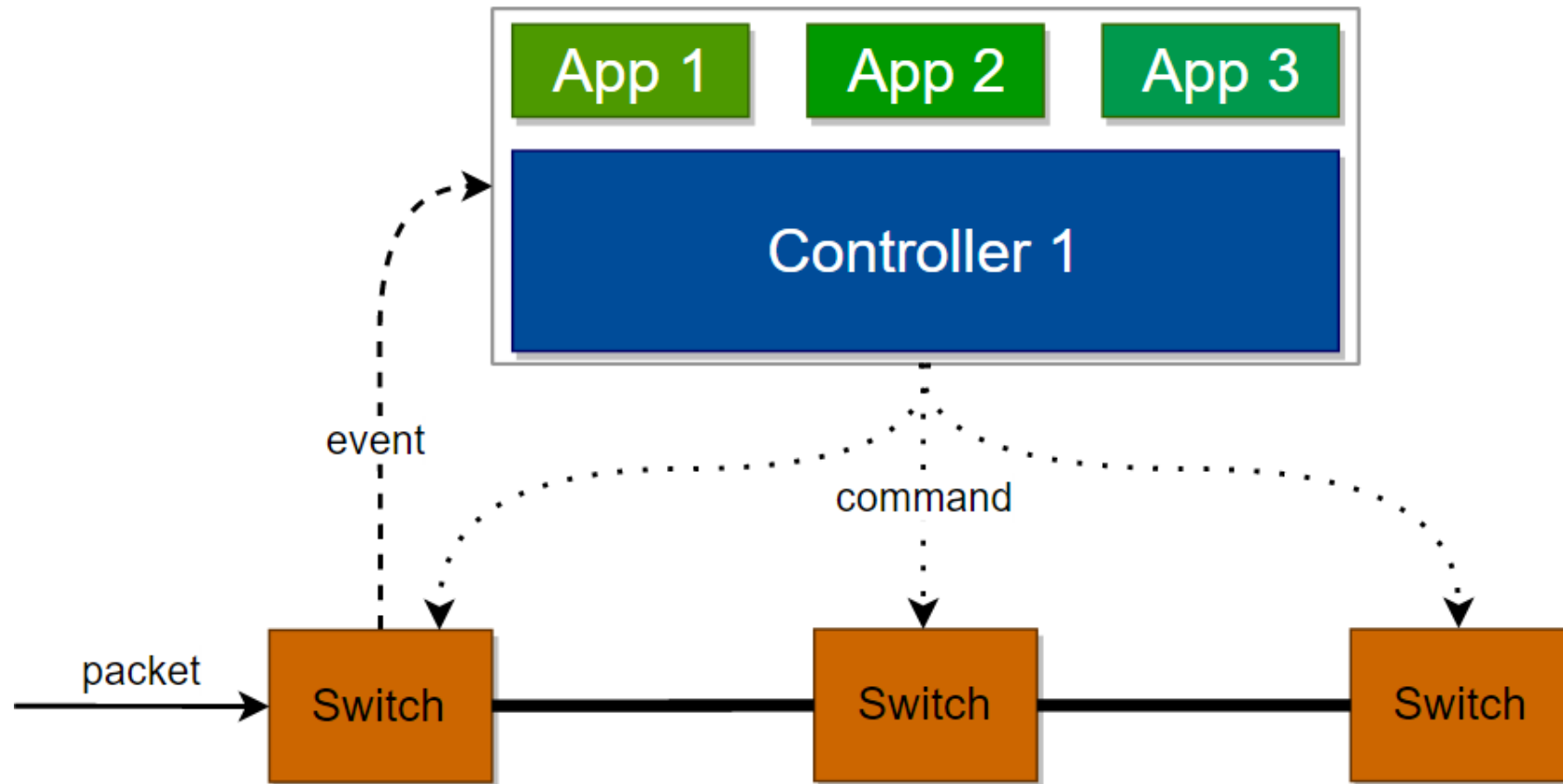
Context



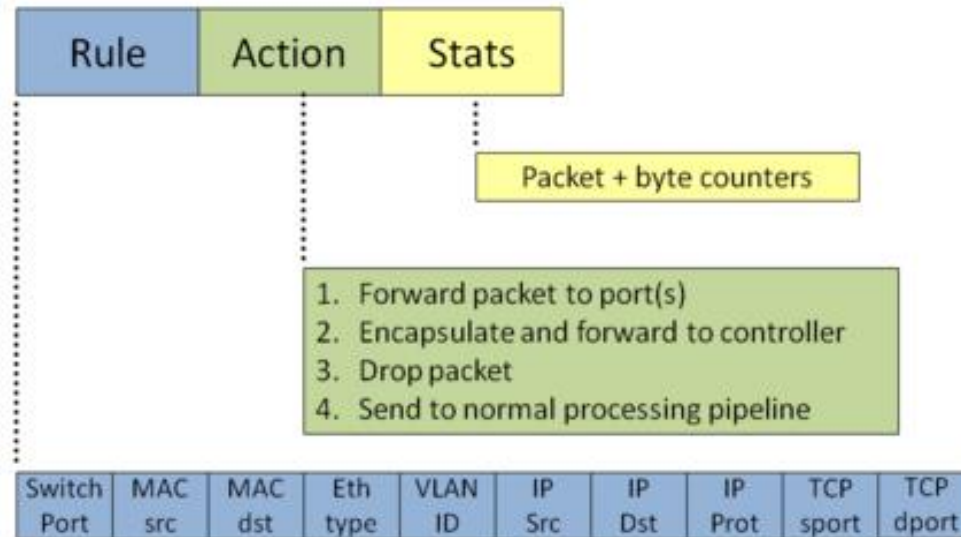
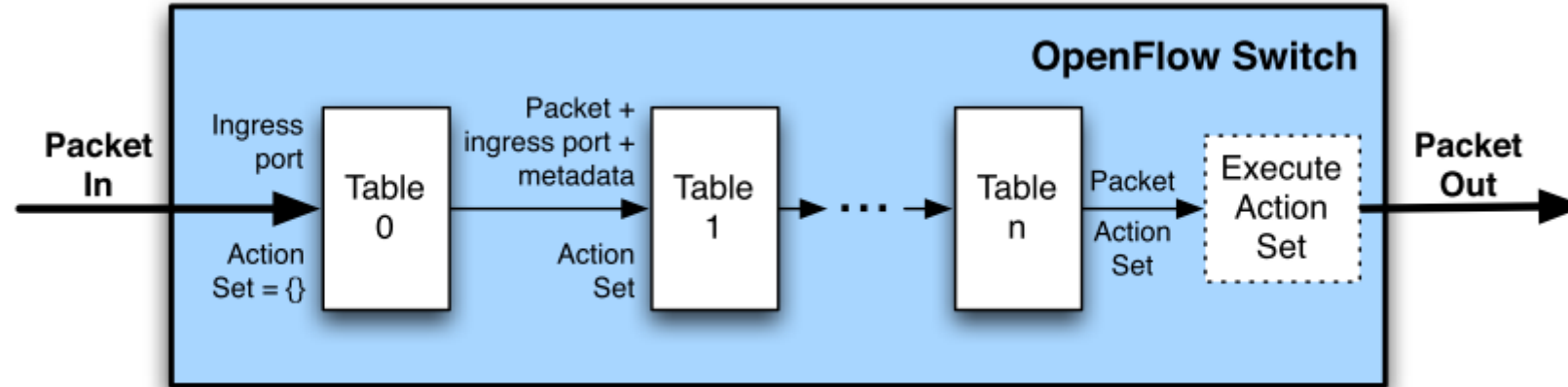
Context



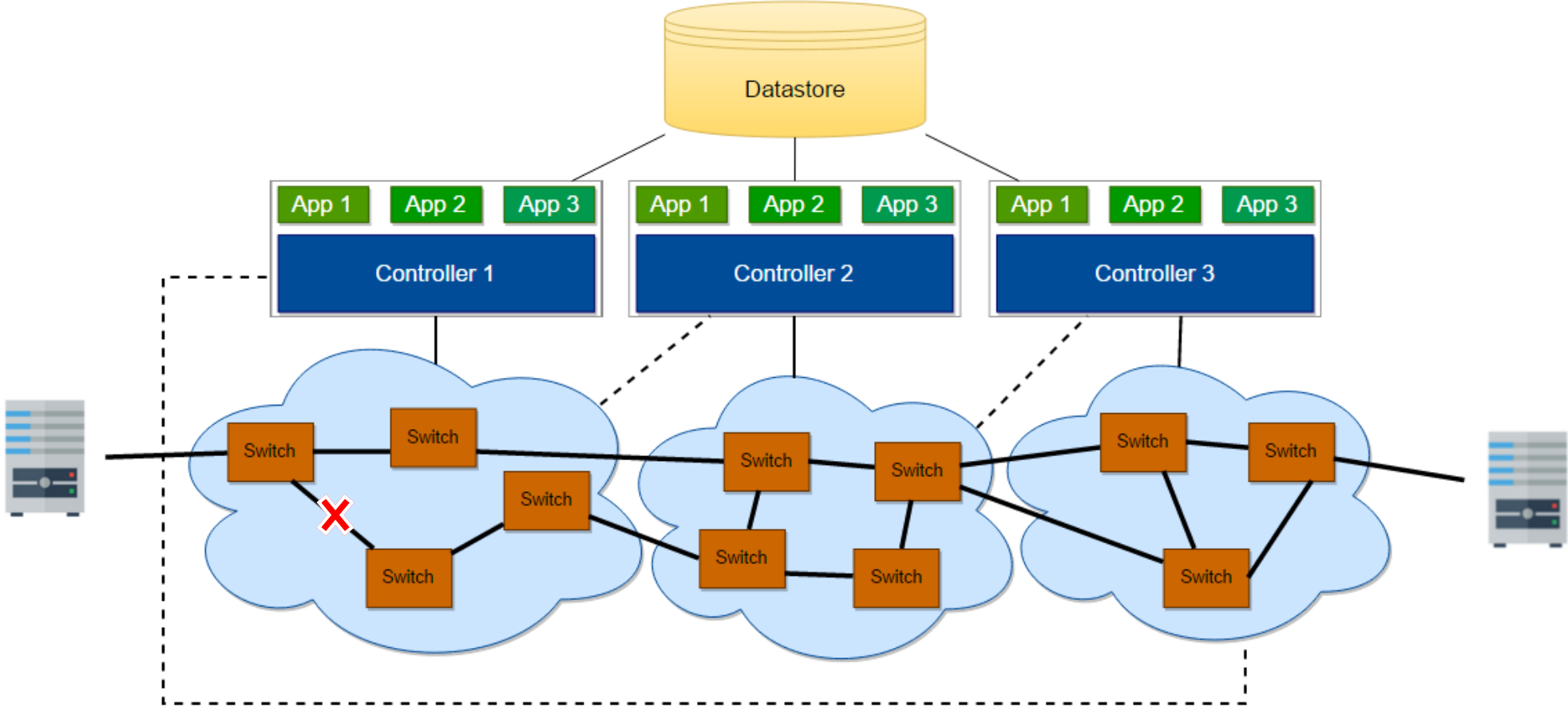
Introduction to SDN



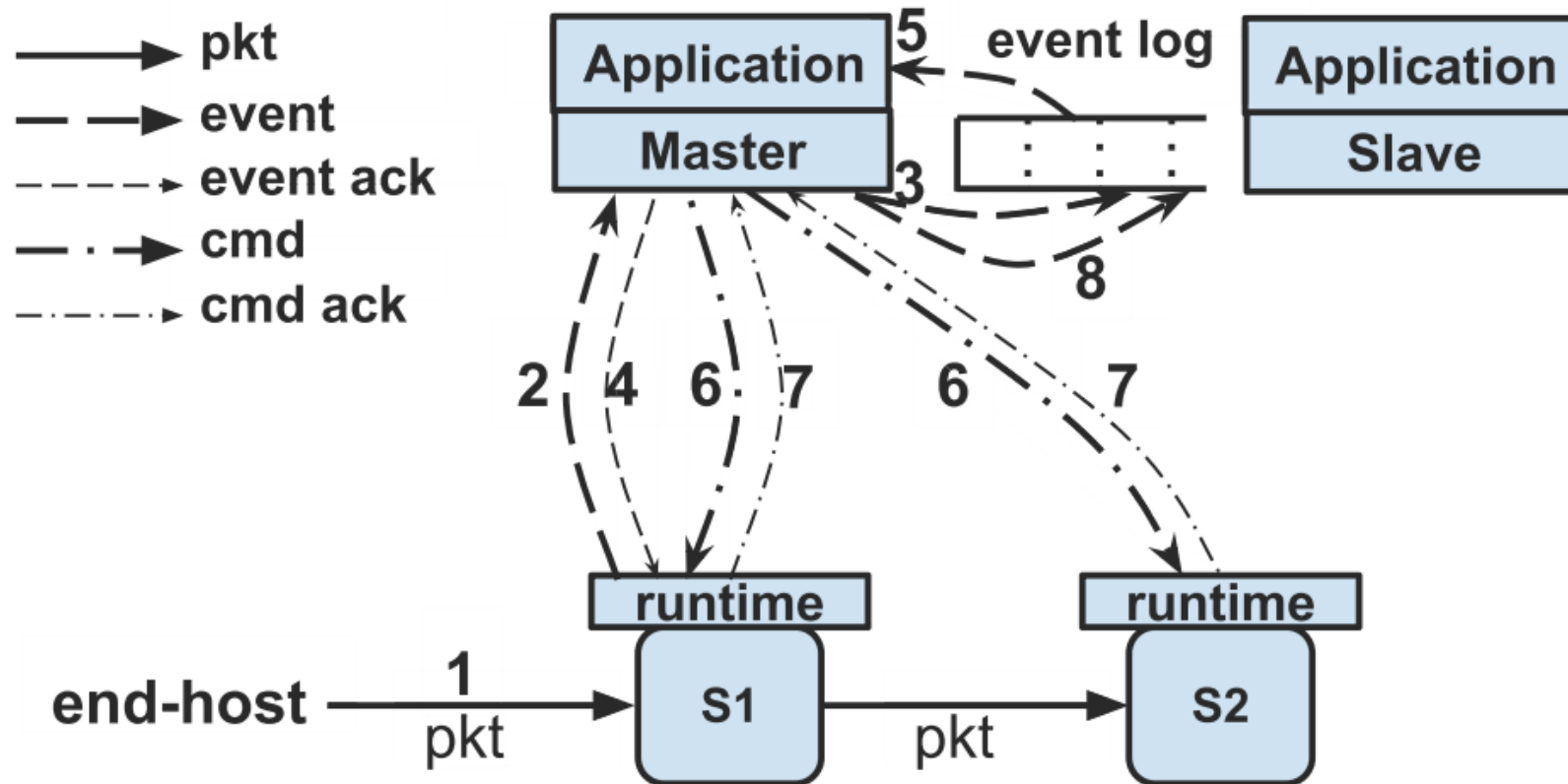
OpenFlow switch



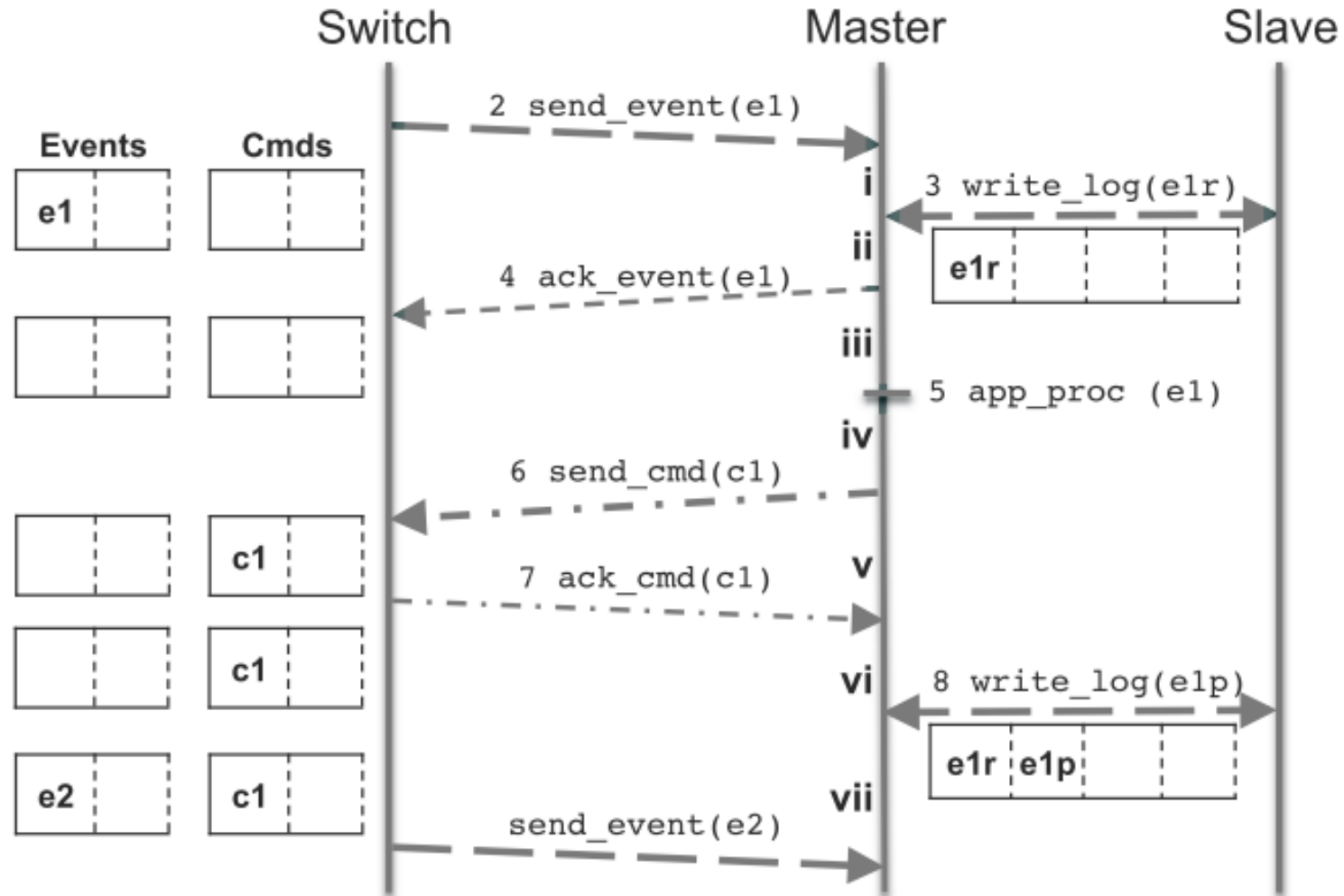
Motivation: Exactly once semantics



Related Work: Ravana



Related Work: Ravana



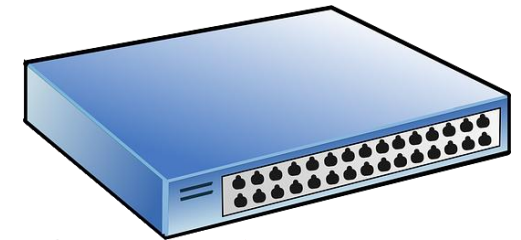
Summing up...

Exactly once events

- Buffering and retransmission
- IDs and filtering in the log

Exactly once commands

- Buffering and filtering by switches
- ACK from switches



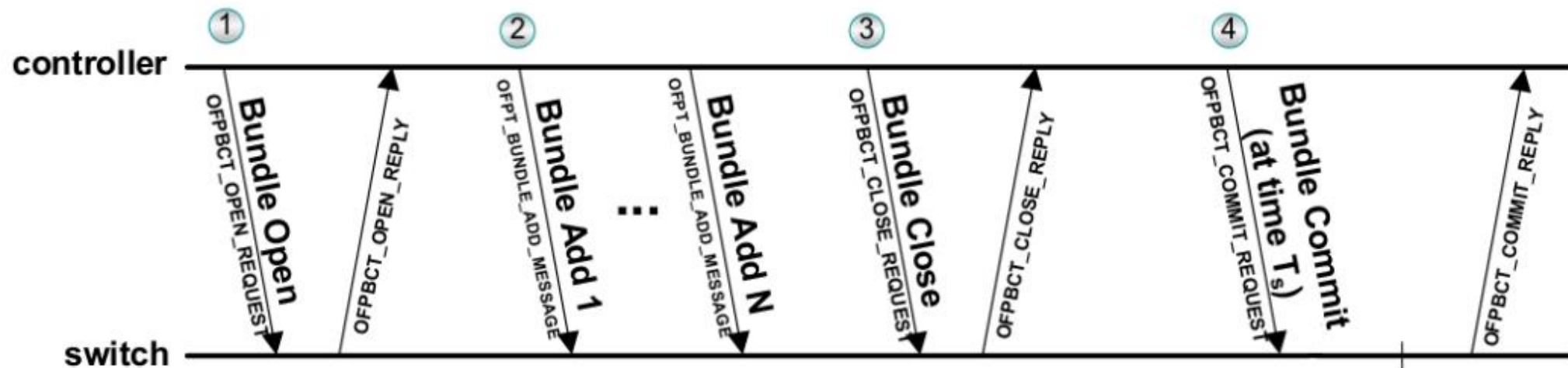
Problem!



Proposal

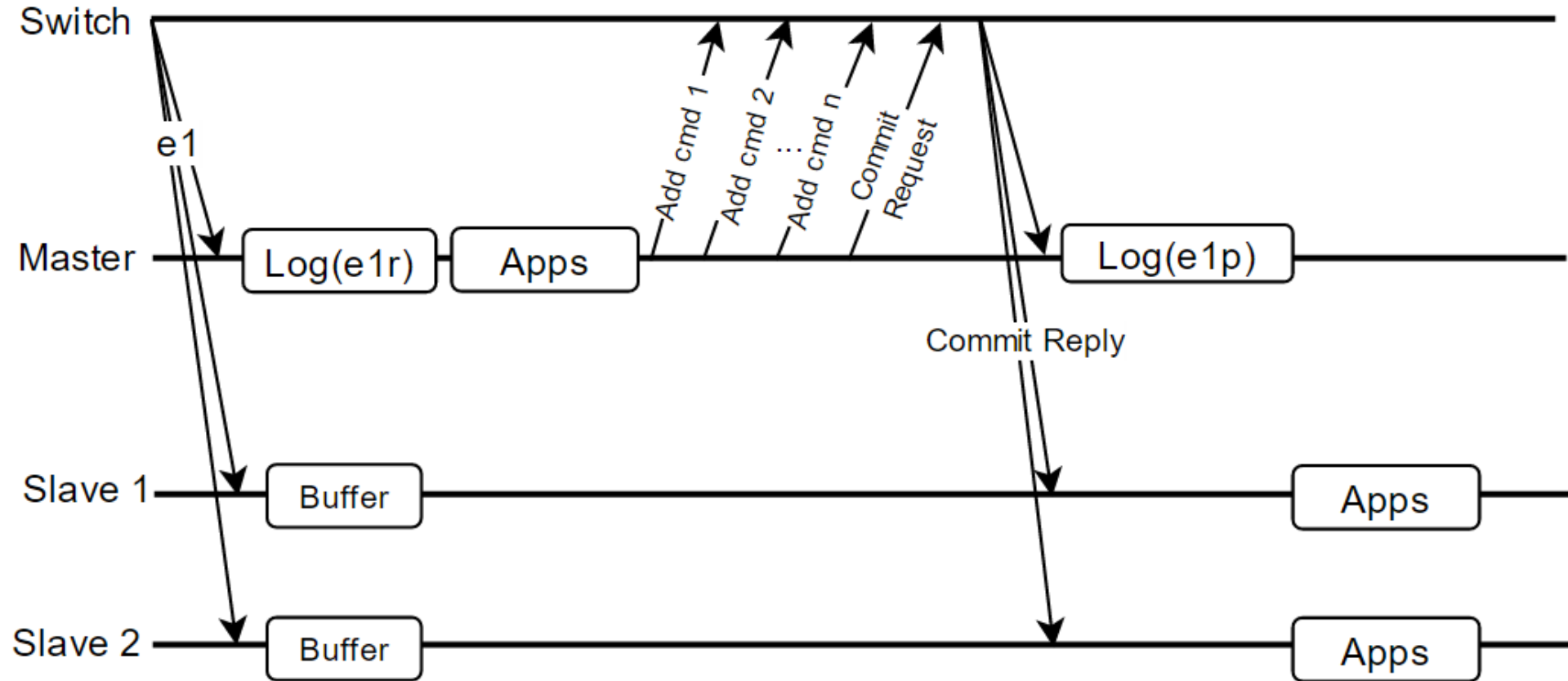
- Distributed architecture: scale with the number of switches
- Fault-tolerant: correct and consistent operation of controllers and switches
- Transparent for applications
- No changes to OpenFlow or switches

OpenFlow 1.4 Bundles

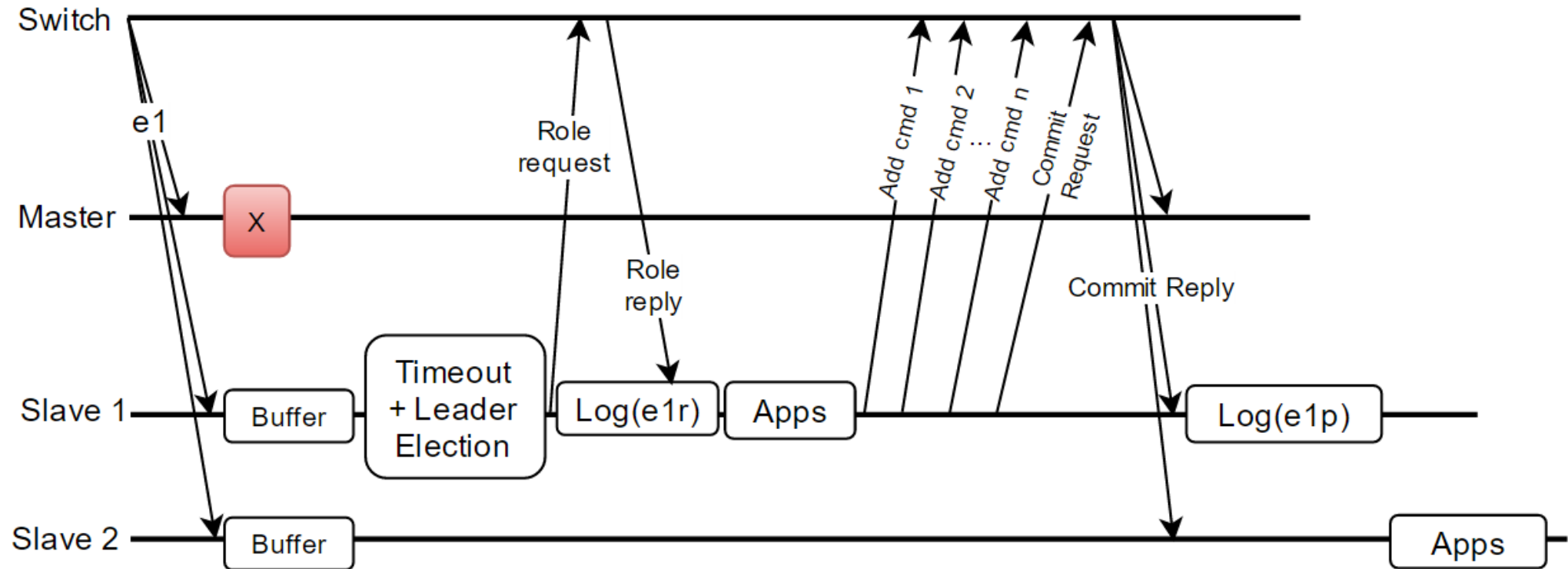


- Available flags: atomic, ordered
- Number of messages can be optimized

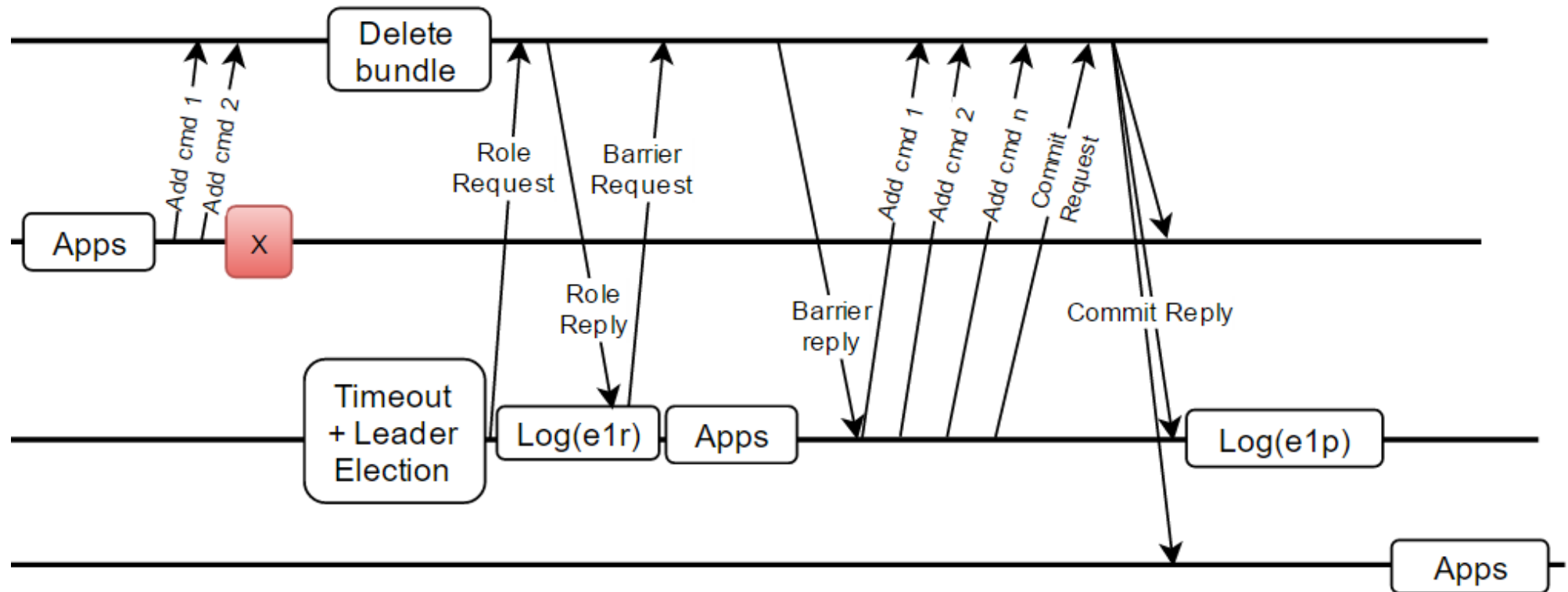
Protocol for Fault-Tolerance



Exactly once events with faults



Exactly once commands with faults



Comparison with Ravana

- Doesn't require changes to switches or OpenFlow
- Distributed architecture scales better
- More messages in the network
- More synchronization between controllers

Plans

- Implementation: Floodlight + BFT-SMaRt
- Evaluation: Cbench
 - Normal case + impact of faults
 - Compare with Ravana
 - Compare with Floodlight

The slide features two large, thick black L-shaped brackets. One is positioned in the top-left corner, and the other is in the bottom-right corner, framing the central text.

Questions?