

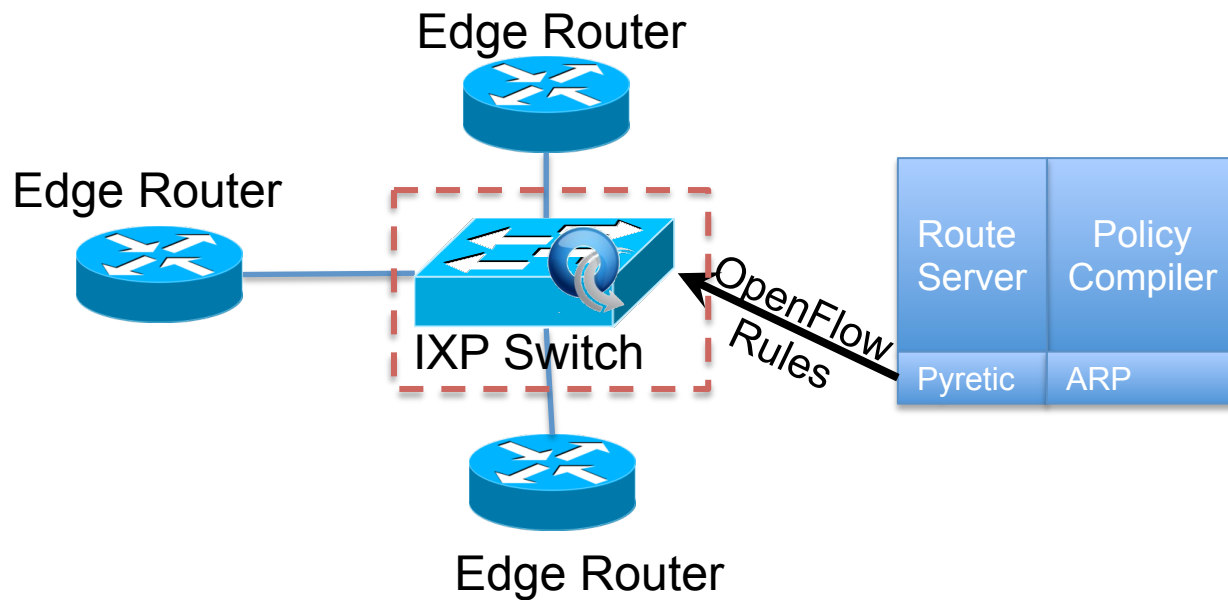
Slick: A control plane for middleboxes

Bilal Anwer, Theophilus Benson, Dave Levin, Nick Feamster, Jennifer Rexford

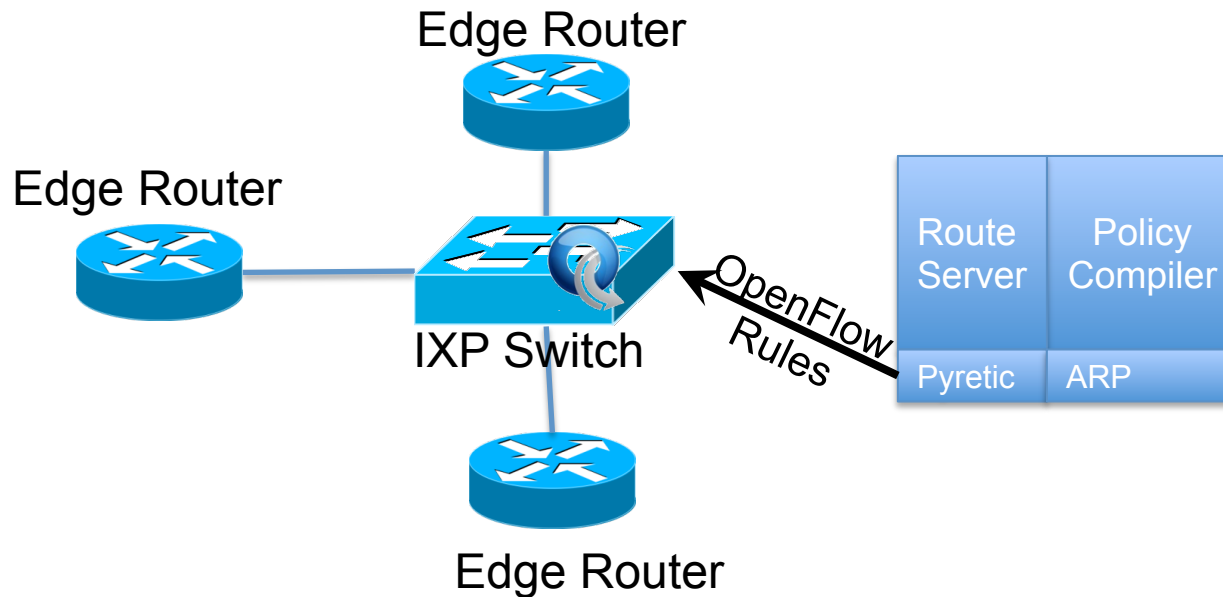
Supported by DARPA through the U.S. Navy SPAWAR under contract N66001-11-C-4017

What is SDX?

- SDX = SDN + IXP
 - An IXP with an SDN data-plane



Limitations of Current SDX Data Plane



SDX Policies:

*Match (**Pattern**), then (**Actions**)*

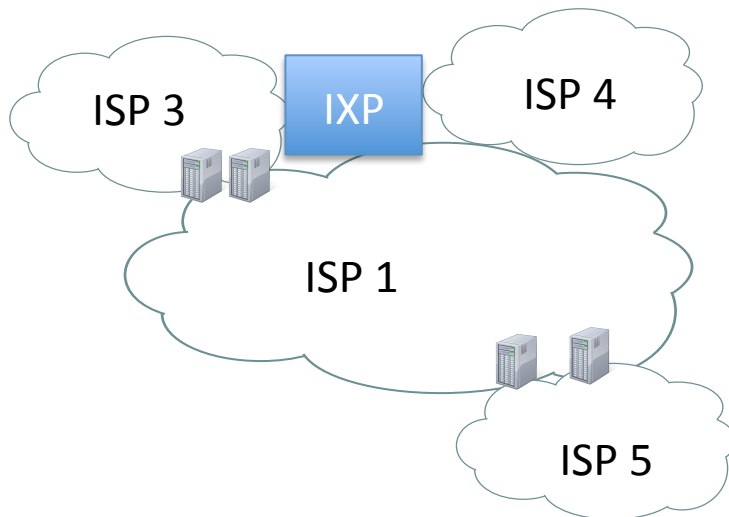
- Limited actions and matching
 - Match: Ethernet, IP, TCP/UDP port numbers
 - Action: forward, drop, rewrite header, etc.

Use Cases for Richer Data Plane Primitives

- Protection against DDoS
 - Match (**ip-prefix**) then (**scrub-traffic**)
 - SDN switches do not support **scrub-traffic** action
 - Requires a middlebox with traffic-scrubbing abilities
- Worm detection
 - Match (**worm-signature**) then (**scrub-traffic**)
 - SDN switches do not support **regular expressions**
 - Requires a middlebox with deep packet inspection abilities

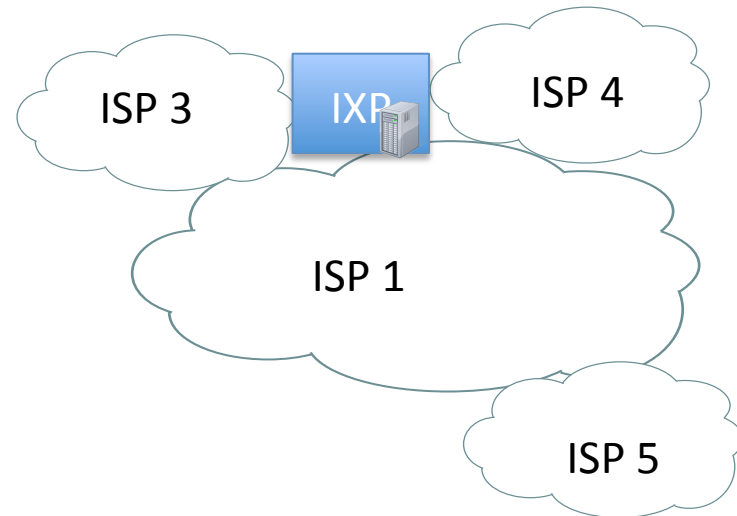
The Case for Middleboxes in IXPs

Middleboxes in ISP's network



- Requires MB at multiple locations
 - Expensive for large ISPs
- Introduces path stretch
 - E.g. traffic from ISP4 must be tunneled to the MB at ISP 5

Middleboxes in IXPs

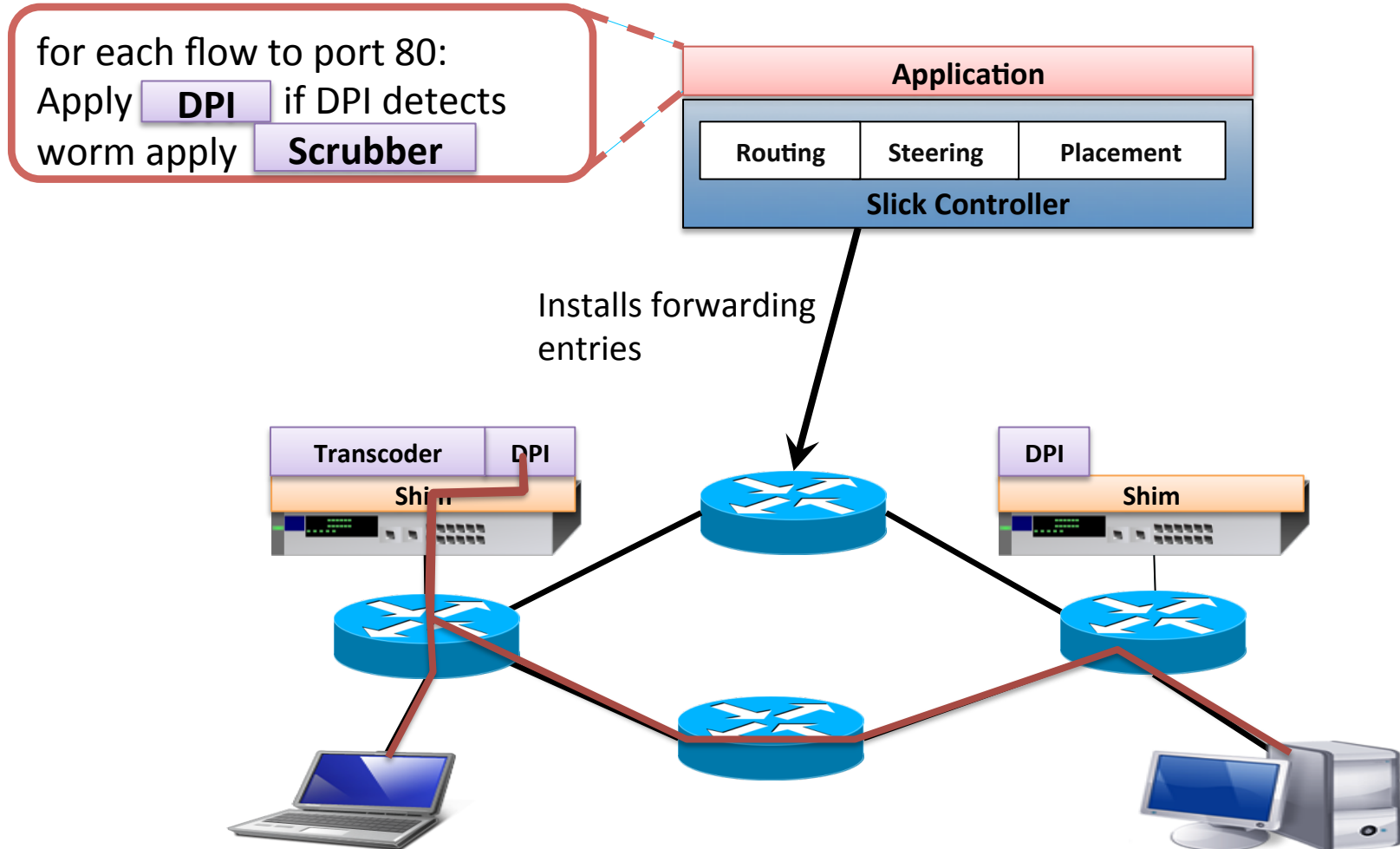


- IXP → single point for multiple peers
- Middleboxes at IXP reduces the total number of middleboxes
- Multiple ISP can share MB

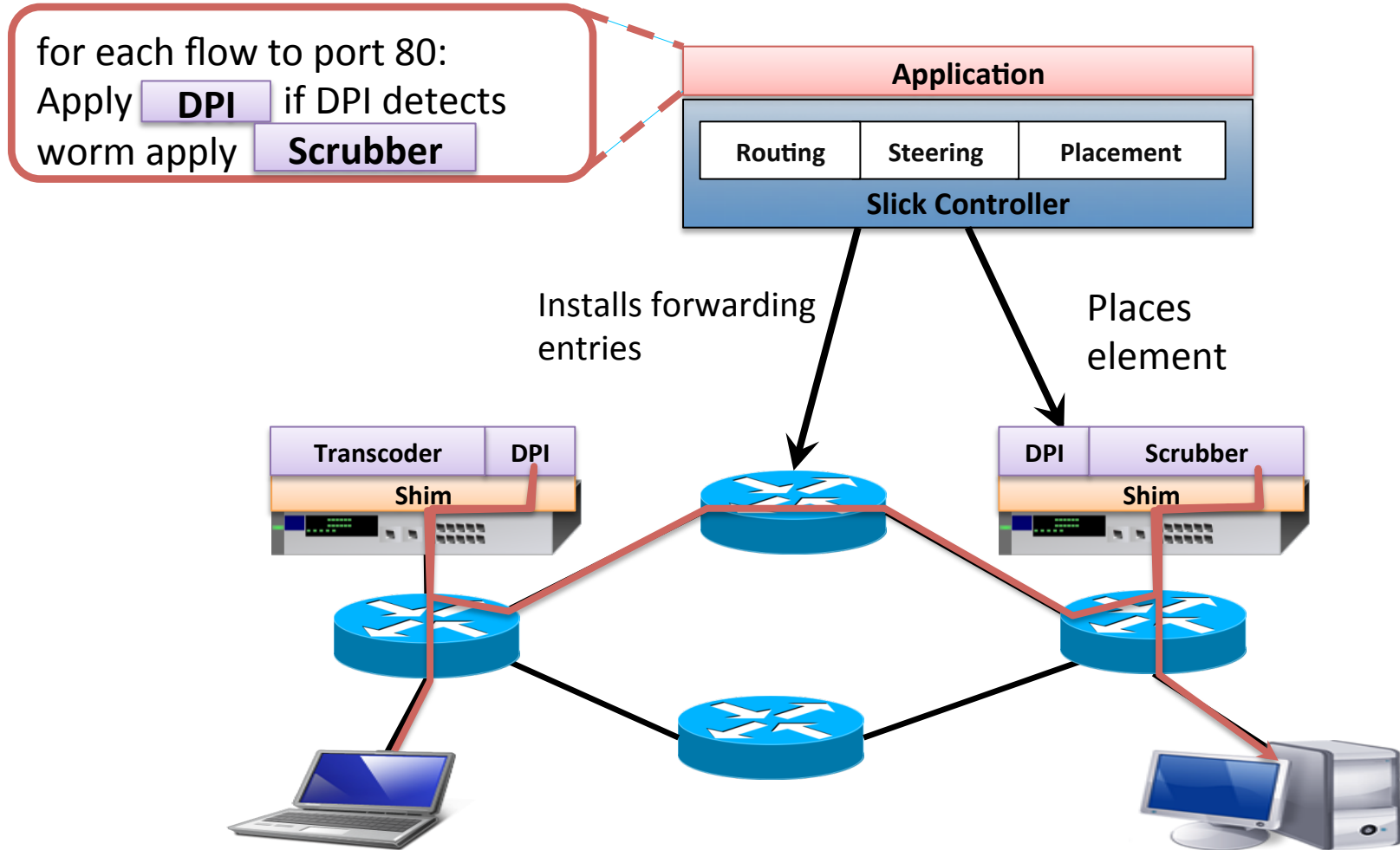
Insight Behind Slick

- Unify control over OpenFlow-based forwarding with middlebox function placement
- Simpler + easier to evolve than alternatives:
 - Expanding the OpenFlow Standard
 - Implementing richer functionality at the controller

Slick Overview

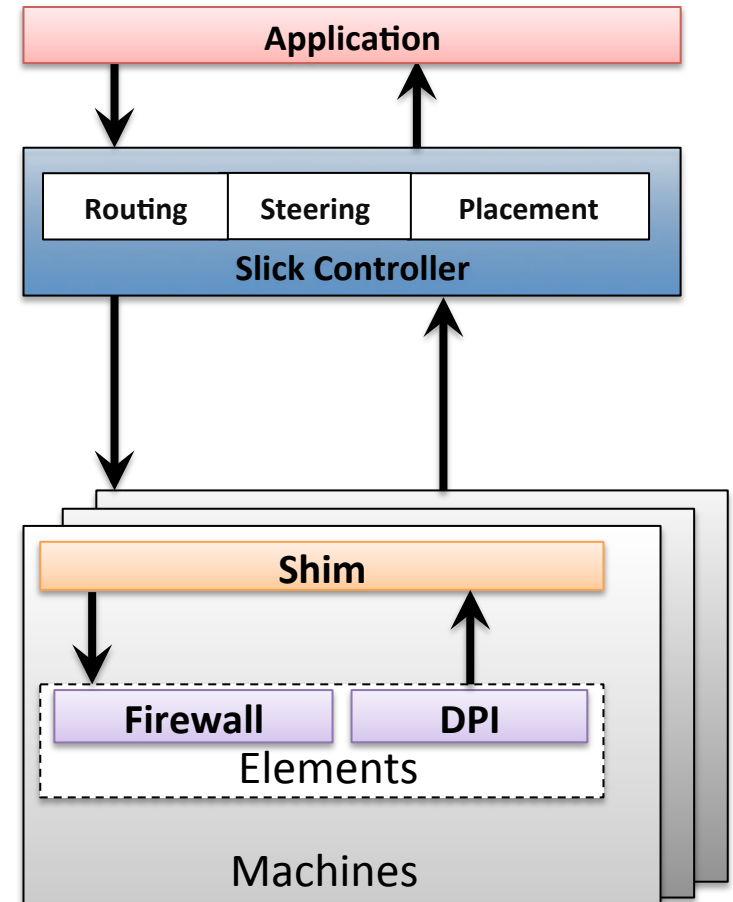


Slick Overview

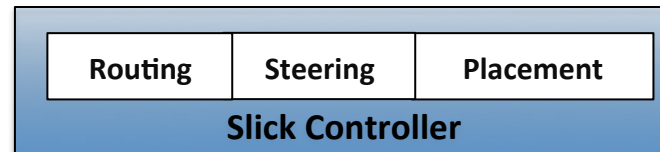


Slick Components

- **Application:**
 - Codification of network policy
- **Controller:**
 - Runs Slick applications
 - Runs online resource allocation algorithm
- **Machine: (Programmable Device)**
 - Runs Slick elements
- **Element:**
 - Modular software code
 - Encapsulates middlebox functionality
- **Shim:**
 - Multiplexes elements on a single machine

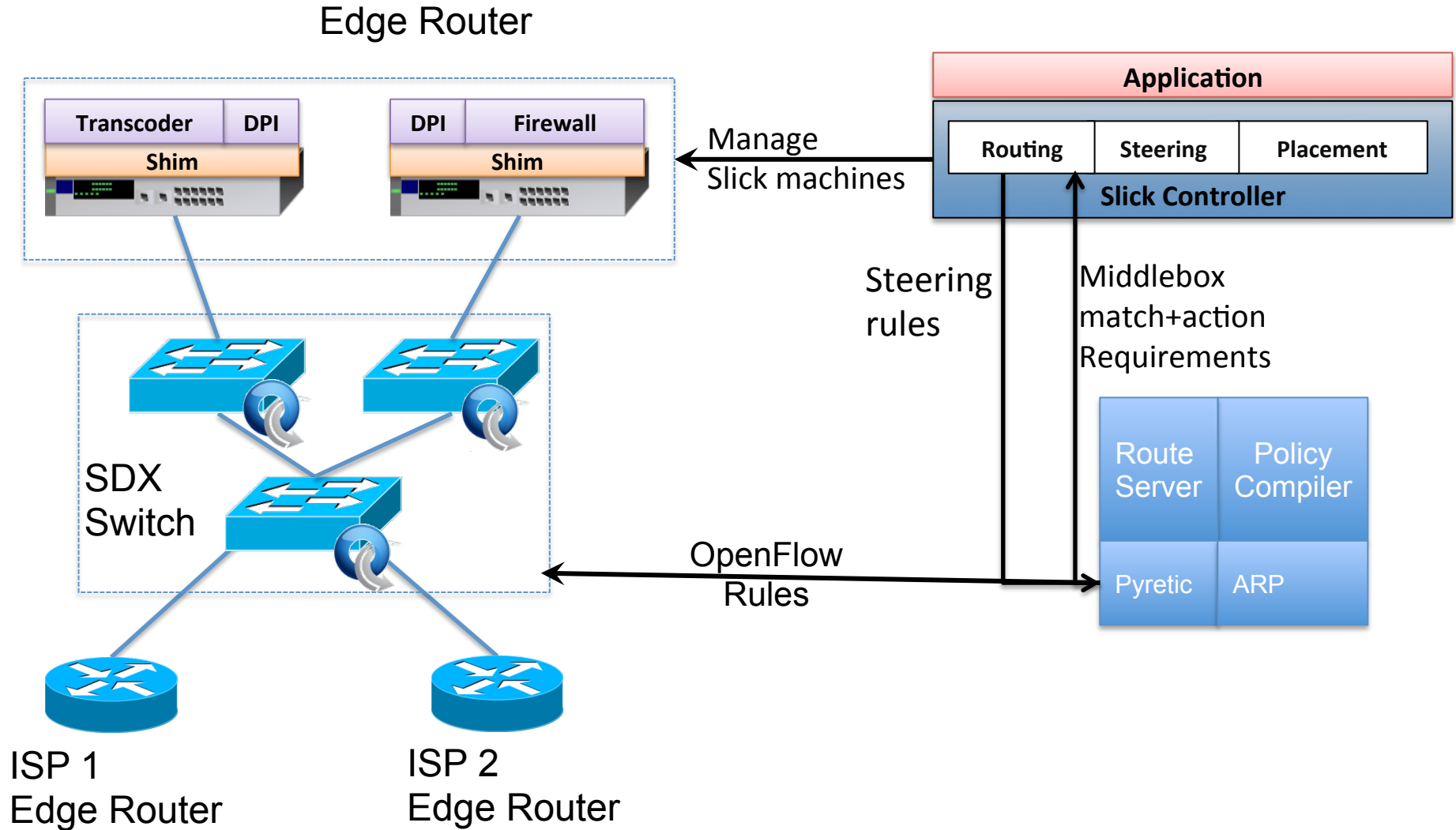


Slick Run Time Algorithm



- Decomposes orchestration problem into 3 sub-problems:
 - Sub-problems operate at different time scales
- Routing (fast): avoids network congestion
 - multi-commodity flow problem
- Traffic Steering (moderate): minimizes stretch
 - Assigns flows to closest element
- Element Placement (slow):
 - Optimization that adjusts placement of elements to changes in traffic demands
 - Determines number of instances of an element
 - Determines the location of each instance

Slick + SDX



Implementation and Current Status

- Slick is implemented in python
 - Slick controller as a module on PoX 0.5.0
 - Developed 3 applications and 5 middlebox elements

Questions?