

SDX Project Updates GEC 20

Georgia Tech Team:
Russ Clark, Nick Feamster, Arpit Gupta
Ron Hutchins, Cas D'Angelo, Siva Jayaraman

June 23, 2014

Project Goals

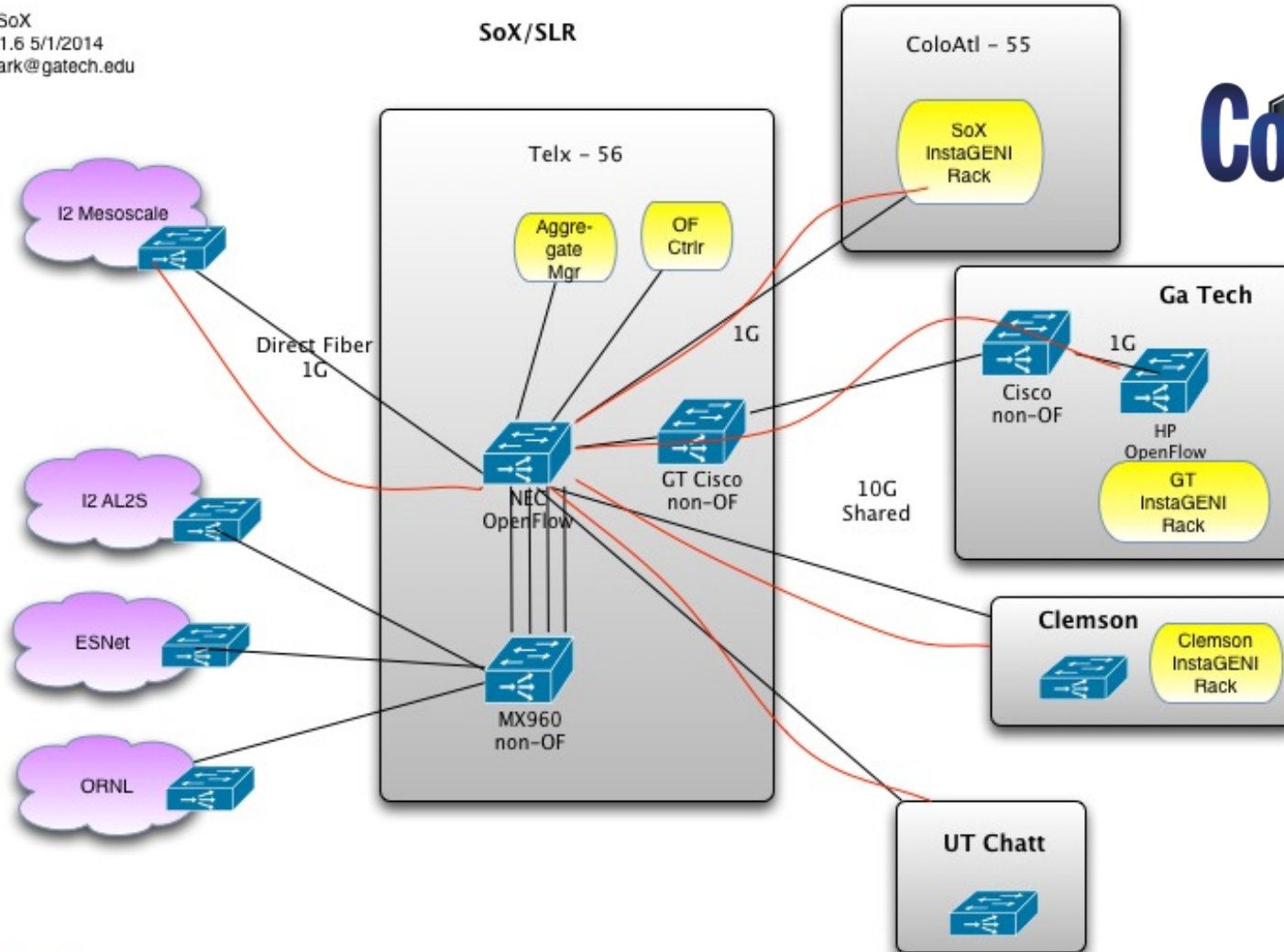
Enable and support SDX research in the GENI community!

- Create a reference architecture
- Provide sample implementations
- Demonstrate and document use in GENI
 - First as a repeatable demonstration experiment
 - Then as a core GENI service

GENI @ SoX



GENI@SoX
Version 1.6 5/1/2014
Russ.Clark@gatech.edu

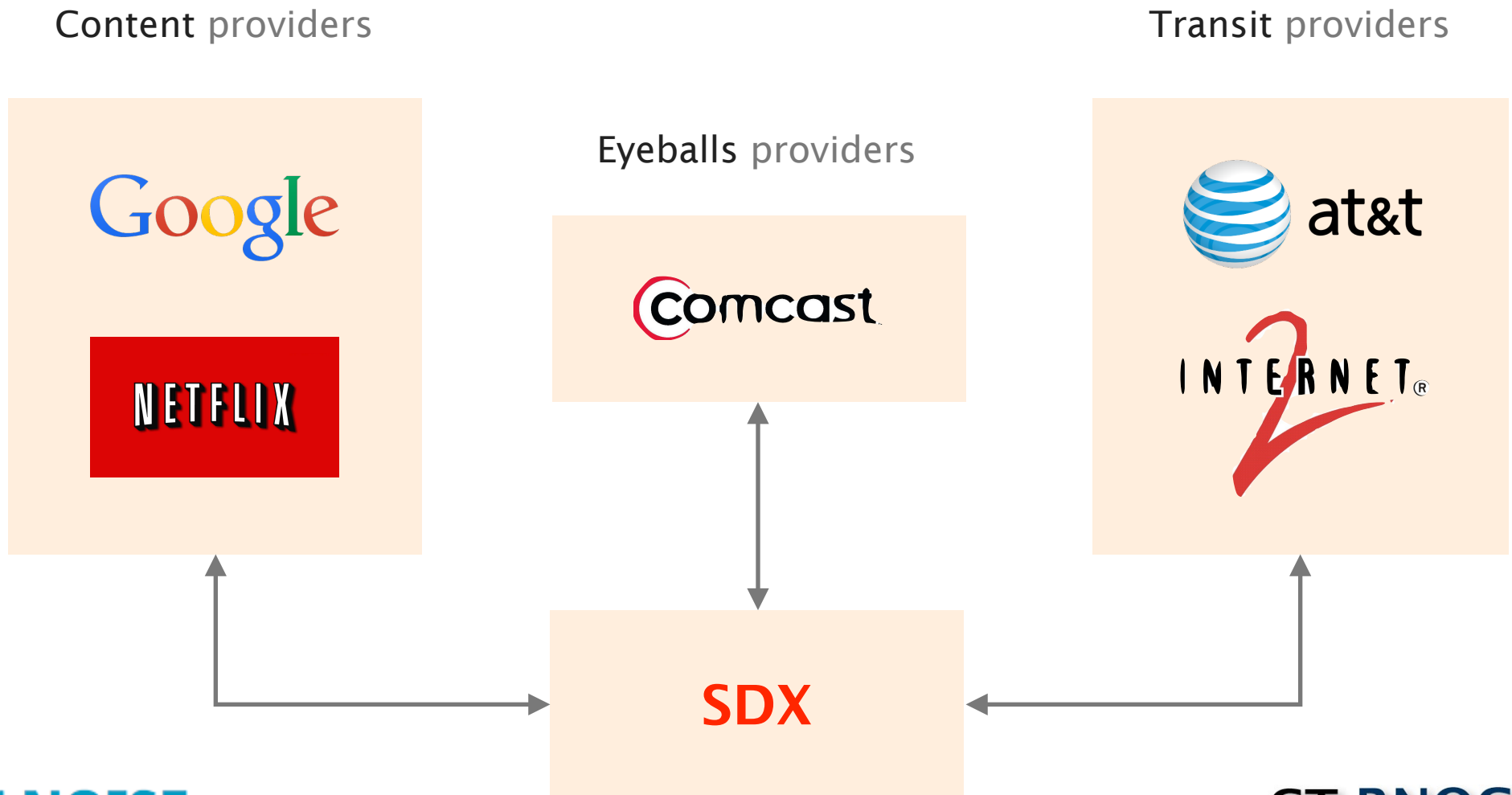


— VLANs 1750 - 1755

What is SDX to us?

- Leverage SDN as a tool to overcome the limitations of traditional peering
 - limitations closely linked to limitations of BGP
 - lack of expressiveness in traffic policy
- Focus on the IXP (> 300 Internet Exchange Points)
- Incrementally deployed as a complementary technology
 - can't replace BGP all at once
- Joint research work with L. Vanbever, M. Shahbaz, S. Donovan, B. Schlinker, N. Feamster, J. Rexford, S. Shenker, R. Clark, E. Katz-Bassett

SDX is a platform that enables multiple stakeholders to define policies/apps over a shared infrastructure



SDX enables a wide range of novel applications

security

Prevent/block policy violation
Prevent participants communication
Upstream blocking of DoS attacks

forwarding optimization

Middlebox traffic steering
Traffic offloading
Inbound Traffic Engineering
Fast convergence

peering

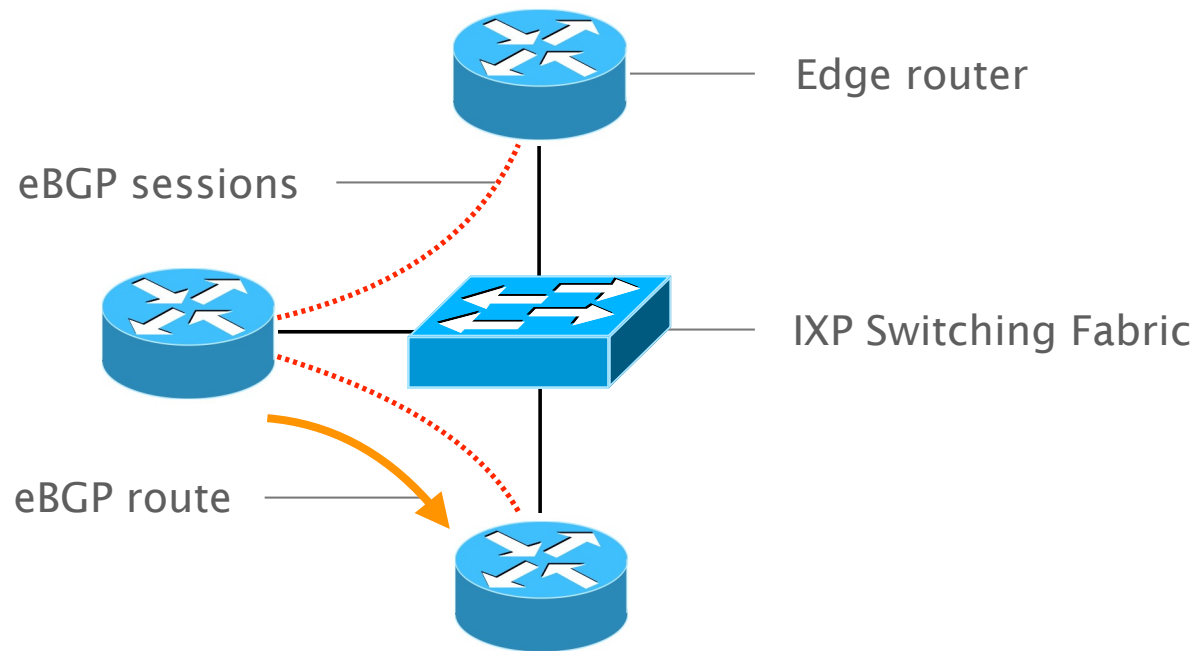
Application-specific peering

remote-control

Influence BGP path selection
Wide-area load balancing

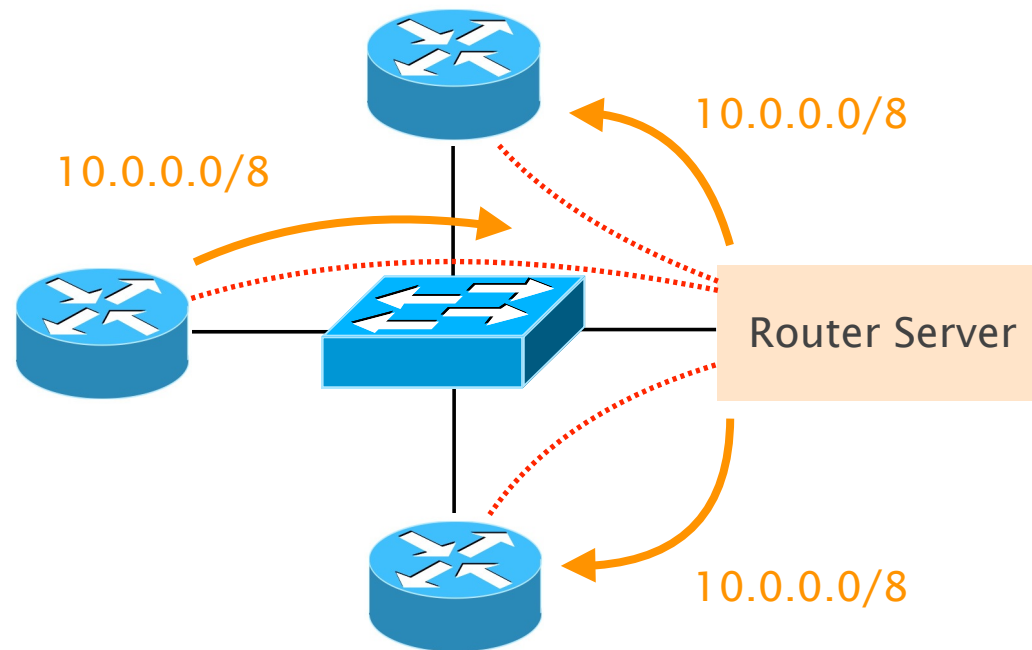
What is an IXP?

An IXP is a large L2 domain where participant routers exchange routes using BGP



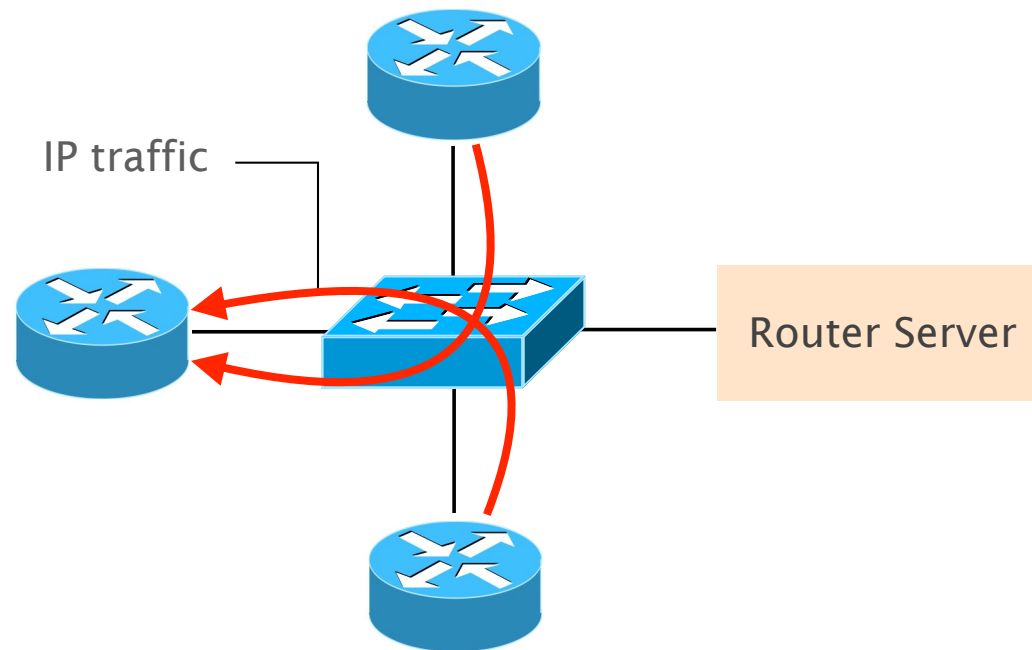
What is an IXP?

To alleviate the need of establishing eBGP sessions, IXP often provides a Route Server (route multiplexer)



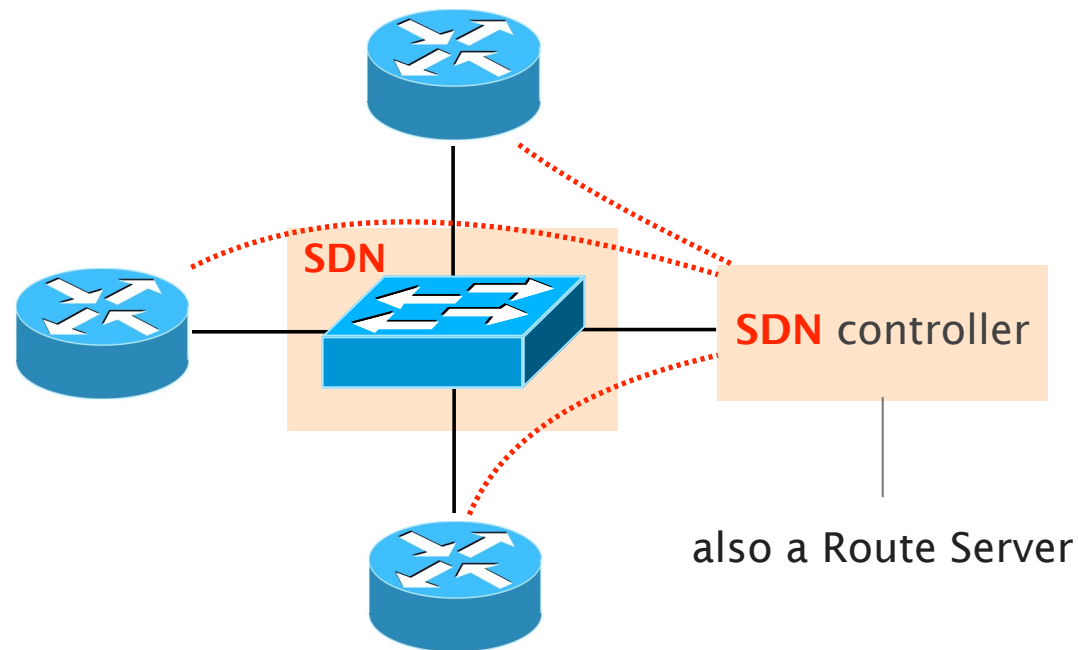
What is an IXP?

IP traffic is exchanged directly between participants,
i.e. the IXP is forwarding transparent



What is an SDX?

With respect to a traditional IXP, SDN-enabled IXP (SDX) control-plane relies on a SDN controller



Programming Abstractions

Abstractions are the key!

Multiple participants share the control plane.

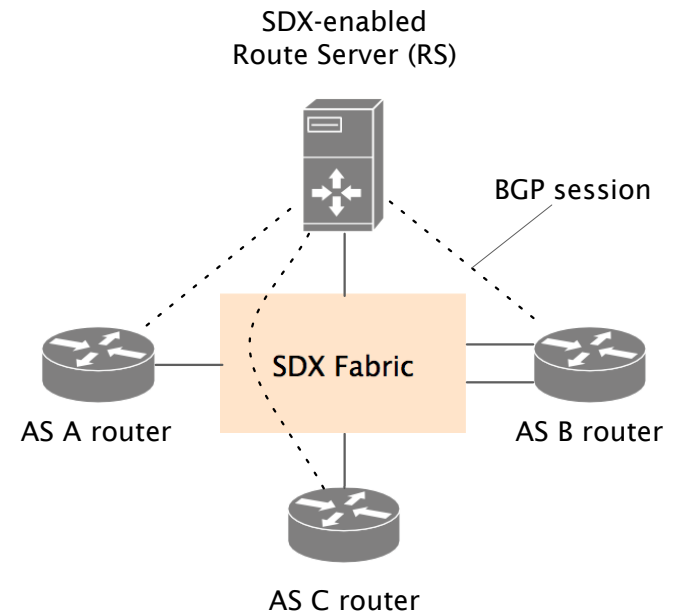
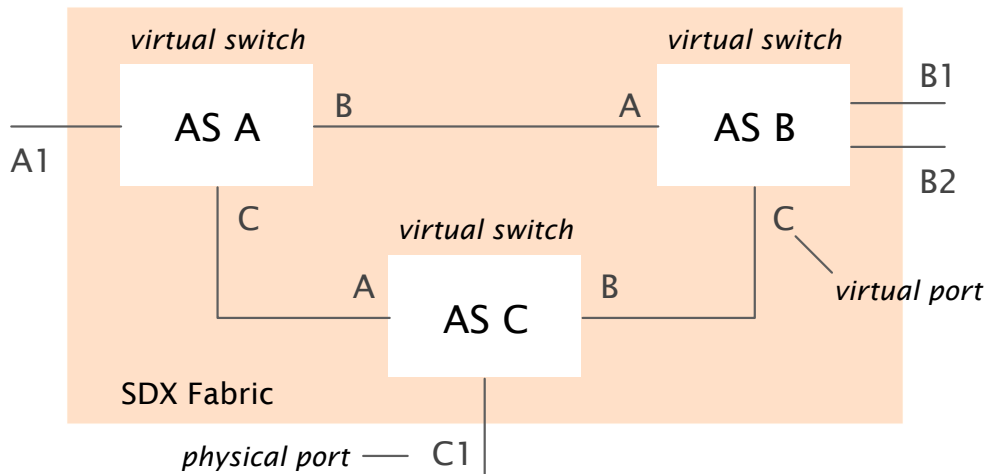
Need right abstractions for:

- Flexibility
- Isolation

SDX policies interact with legacy protocols eg. BGP. Need right abstractions to ensure compatibility and correctness

Virtual SDX Switch Abstraction

Be your own boss!



Illusion of its own virtual SDX switch for each participant

Balances desire for flexibility with necessary isolation

Virtual SDX Switch Abstraction

How do participants write policy? **Pyretic**

Pattern

```
match ( eth_type  
        vlan_id  
        srcmac  
        dstmac , && , || ) , then (  
        protocol  
        dstip  
        tos  
        srcip  
        srcport  
        dstport
```

Actions

```
drop  
forward  
rewrite
```

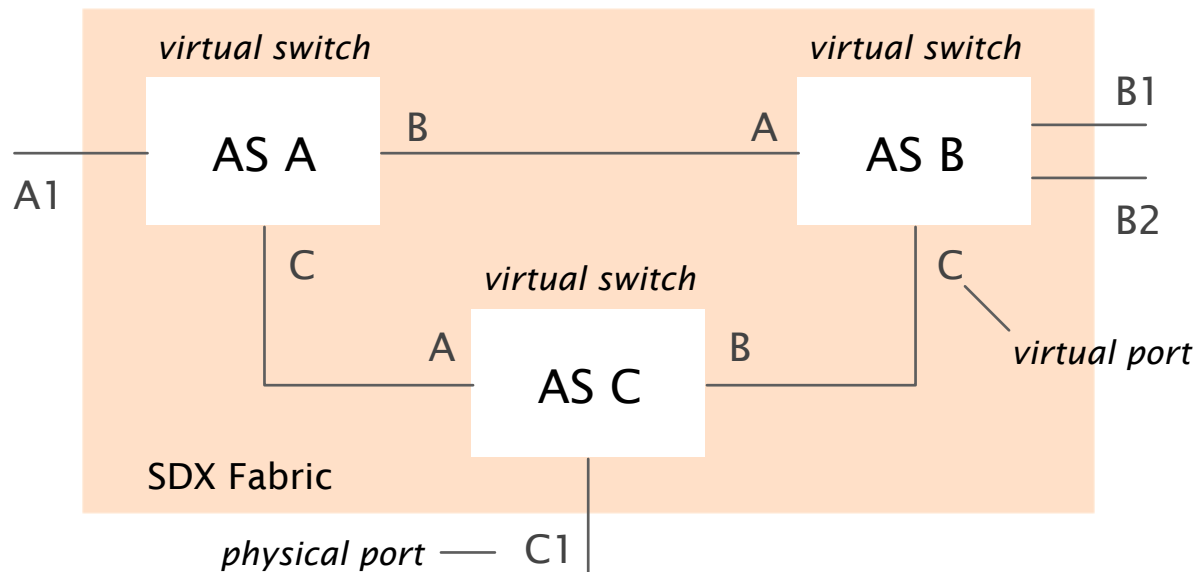
Virtual SDX Switch Abstraction

AS A policy: application-specific peering

```
(match(dstport=80) >> fwd(B)) +
(match(dstport=443) >> fwd(C))
```

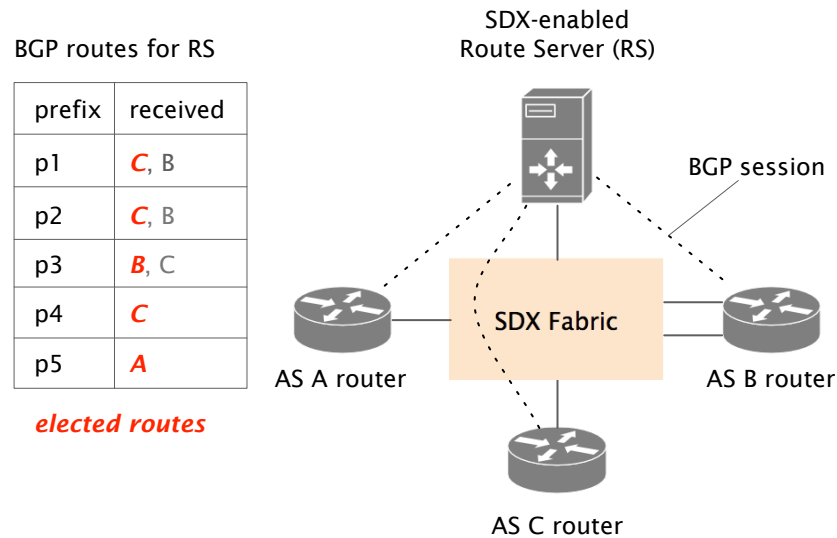
AS B policy: inbound traffic engineering

```
(match(srcip={0/1}) >> fwd(B1)) +
(match(srcip={128/1}) >> fwd(B2))
```



Integration with Inter-domain Routing

Forwarding policies apply only for prefixes announced by neighbor



`match(dstport=80) >> fwd(B)`

`match(dstport=443) >> fwd(C)`

Applied only for prefixes

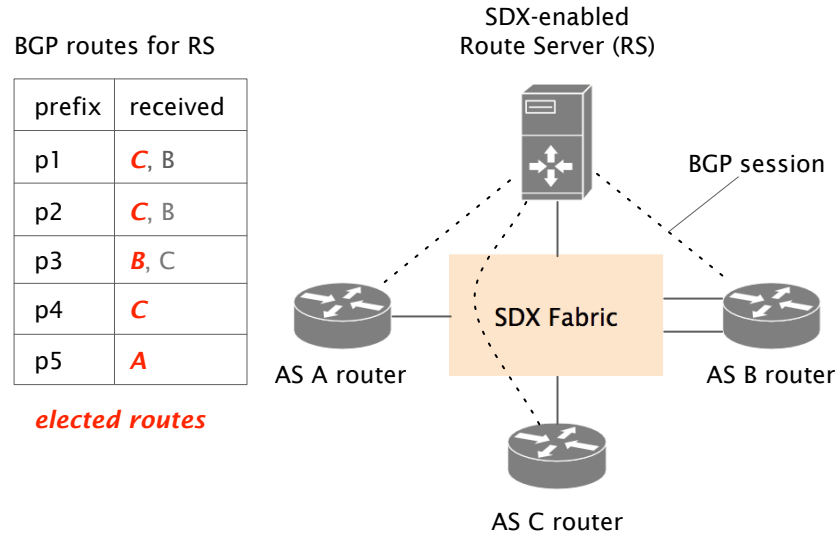
p1, p2, p3

Applied only for prefixes

p1, p2, p3, p4

Integration with Inter-domain Routing

Non-matching traffic follows default BGP forwarding.



What happens with traffic for dstip=p2 dstport=22?

`match(dstport=80) >> fwd(B)`

`match(dstport=443) >> fwd(C)`

Applied only for prefixes

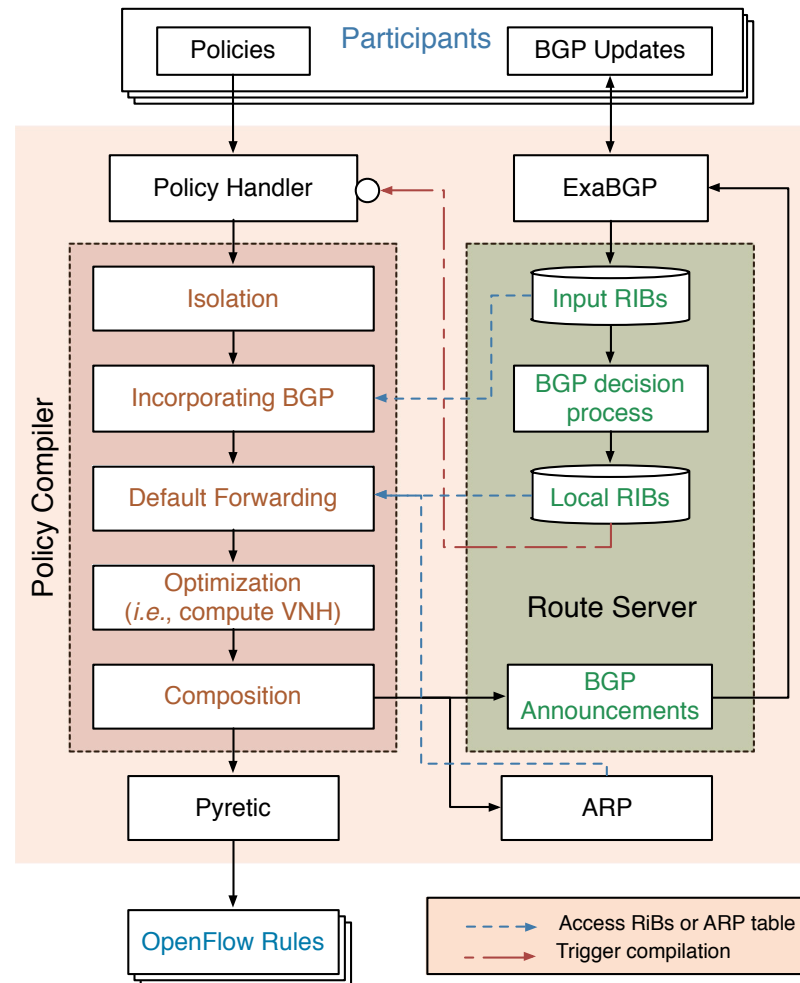
p1, p2, p3

Applied only for prefixes

p1, p2, p3, p4

Controller Design

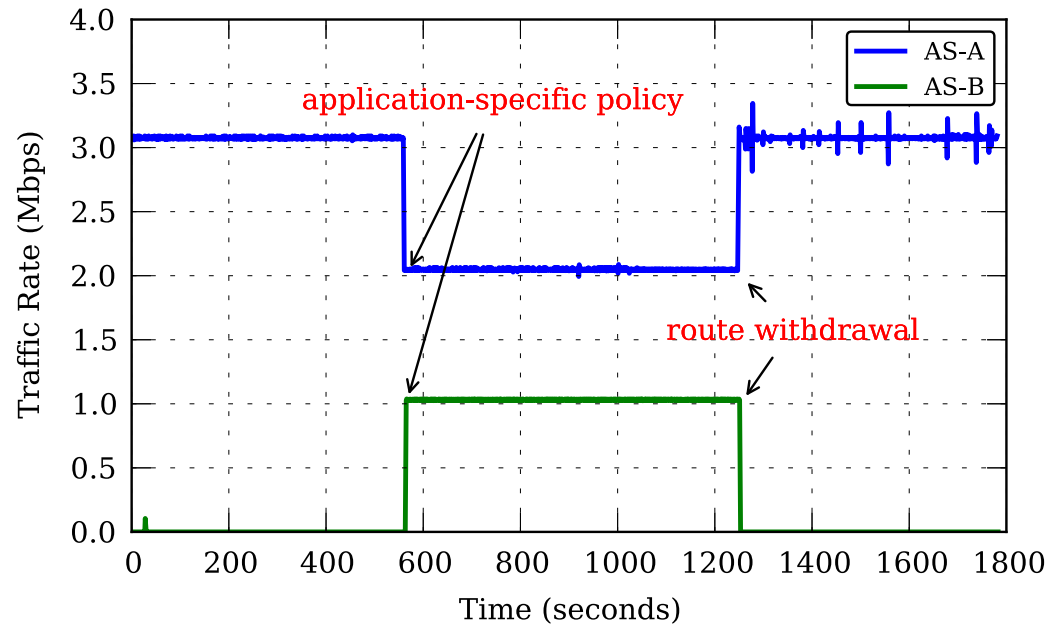
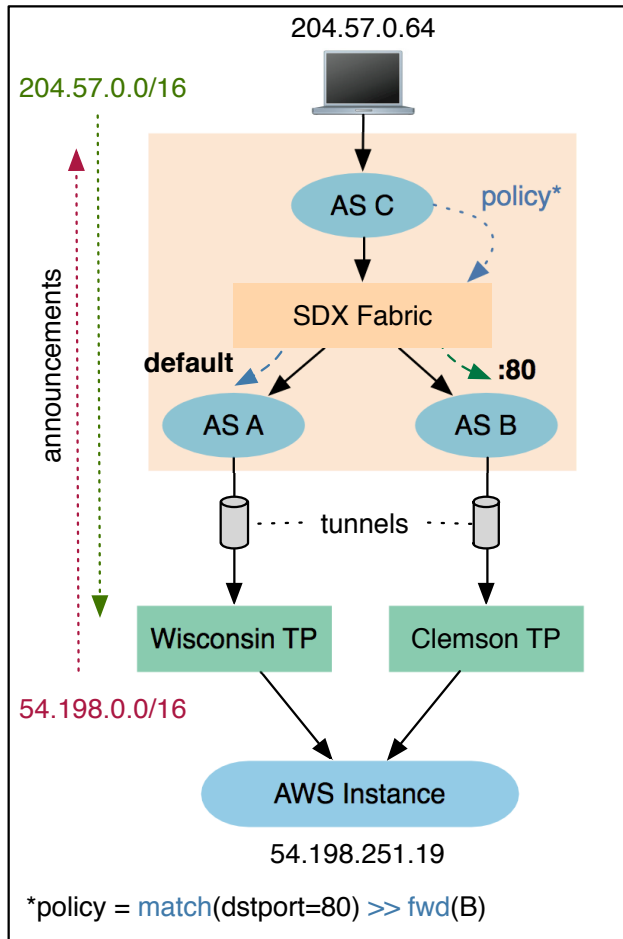
What does the SDX controller look like?



How do I use it?

- Deploy a controller host
 - operates the SDX controller software and route server
- Create your network topology
- Point your switches at the controller
- Define policy rules using policy language
- See example on GitHub
 - <https://github.com/sdn-ixp/sdx-platform/wiki/>

Application Specific Peering



Going Forward

- Deploy current SDX controller as a GENI demonstration project
- Get people using it!
- Iterate, refine, improve.
- Work to deploy as a core service with peering arrangements
 - build on GEC 19 demo
 - R&E and commercial

Contact Info

- Russ.Clark@gatech.edu
- feamster@cc.gatech.edu
- agupta80@gatech.edu