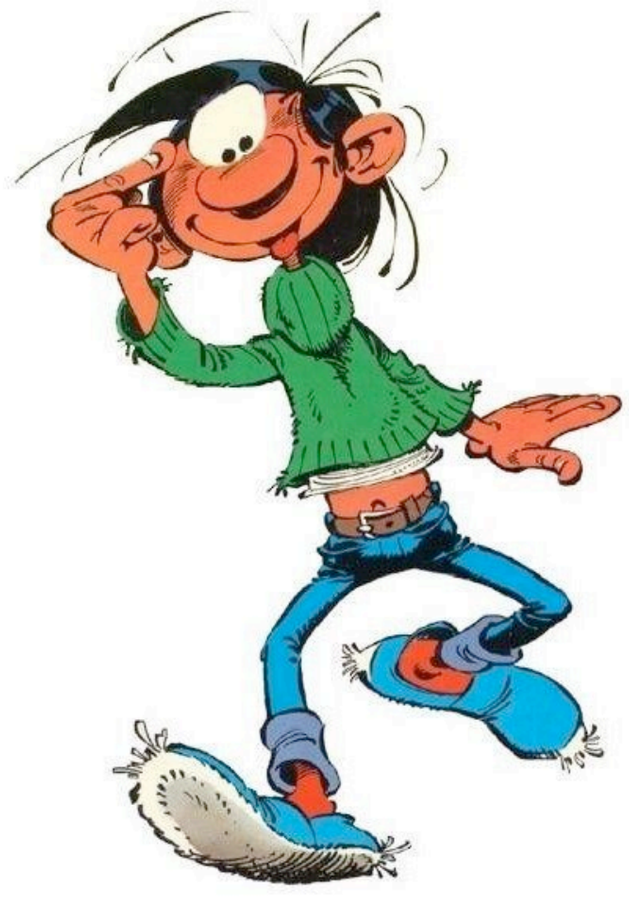
# Novel Applications for a SDN-enabled Internet Exchange Point



Laurent Vanbever

vanbever@cs.princeton.edu

SDN Research Group, IETF87

July, 29 2013

Based on joint work with

Arpit Gupta, Muhammad Shahbaz, Hyojoon Kim,
Russ Clark, Nick Feamster, Jennifer Rexford and Scott Shenker

# BGP is notoriously unflexible and difficult to manage

Operating BGP has at least three limitations

# BGP is notoriously unflexible and difficult to manage

Operating BGP has at least three limitations

- assume destination IP based routing

# BGP is notoriously unflexible and difficult to manage

Operating BGP has at least three limitations ***what people really want***

- assume destination IP based routing — customized routing decisions

# BGP is notoriously unflexible and difficult to manage

Operating BGP has at least three limitations ***what people really want***

- assume destination IP based routing customized routing decisions
- policies are applied to direct neighbors

# BGP is notoriously unflexible and difficult to manage

Operating BGP has at least three limitations — ***what people really want***

- assume destination IP based routing — customized routing decisions
- policies are applied to direct neighbors — affect end-to-end paths

# BGP is notoriously unflexible and difficult to manage

| Operating BGP has at least three limitations | ***what people really want*** |
| --- | --- |
| assume destination IP based routing | customized routing decisions |
| policies are applied to direct neighbors | affect end-to-end paths |
| indirectly influence forwarding paths | |

# BGP is notoriously unflexible and difficult to manage

| Operating BGP has at least three limitations | ***what people really want*** |
| --- | --- |
| assume destination IP based routing | customized routing decisions |
| policies are applied to direct neighbors | affect end-to-end paths |
| indirectly influence forwarding paths | directing traffic on specific paths |

# SDN can enable fine-grained, flexible and direct expression of interdomain policies

SDN devices forward based on any packet-header fields

at line rate, enabling flexible forwarding

SDN controller can be controlled by remote parties

on a bilateral basis, without any global standards

SDN controller exerts direct control on the data plane

using a standardized API such as OpenFlow

# Internet Exchange Points are perfect places to deploy new interdomain features

# Internet Exchange Points are perfect places to deploy new interdomain features

Internet Exchange Points (IXPs) ...

# Internet Exchange Points are perfect places to deploy new interdomain features

Internet Exchange Points (IXPs)

- connect a large number of participants

# Internet Exchange Points are perfect places to deploy new interdomain features

Internet Exchange Points (IXPs) — AMS-IX:

- connect a large number of participants — 600 participants

# Internet Exchange Points are perfect places to deploy new interdomain features

Internet Exchange Points (IXPs) — AMS-IX:

- connect a large number of participants — 600 participants
- carry a large amount of traffic — > 2250 Gb/s (peak)

# Internet Exchange Points are perfect places to deploy new interdomain features

Internet Exchange Points (IXPs)

- connect a large number of participants
- carry a large amount of traffic
- are a hotbed of innovation

AMS-IX:

- 600 participants
- > 2250 Gb/s (peak)
- BGP Route Server
- Mobile peering
- Open peering
- ...

# Internet Exchange Points are perfect places to deploy new interdomain features

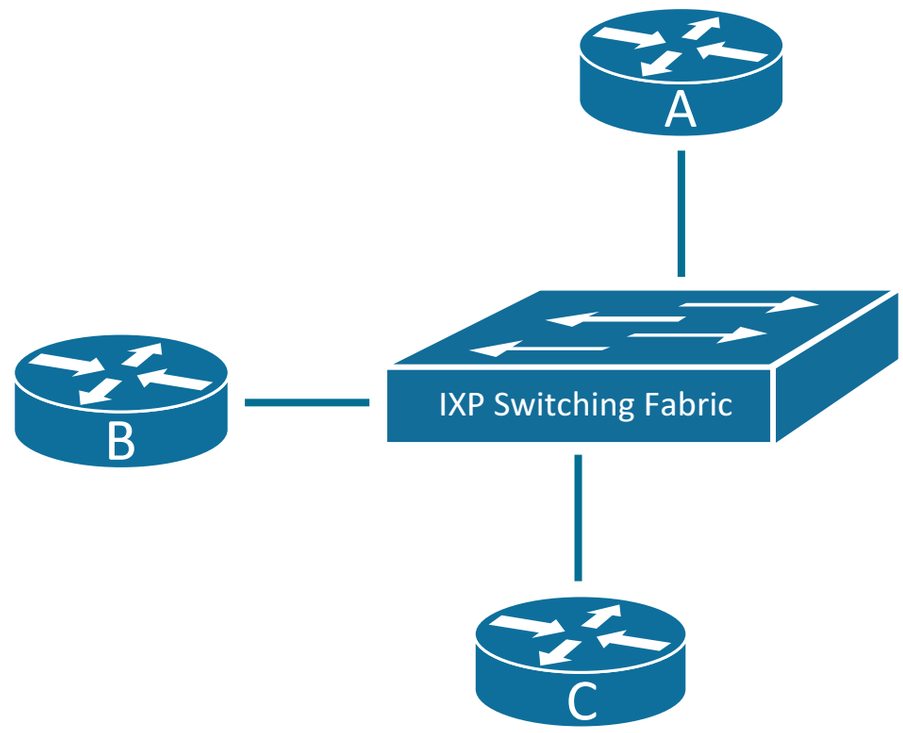
Internet Exchange Points (IXPs)

- connect a large number of participants
- carry a large amount of traffic
- are a hotbed of innovation

Even a single deployment can have a large impact!

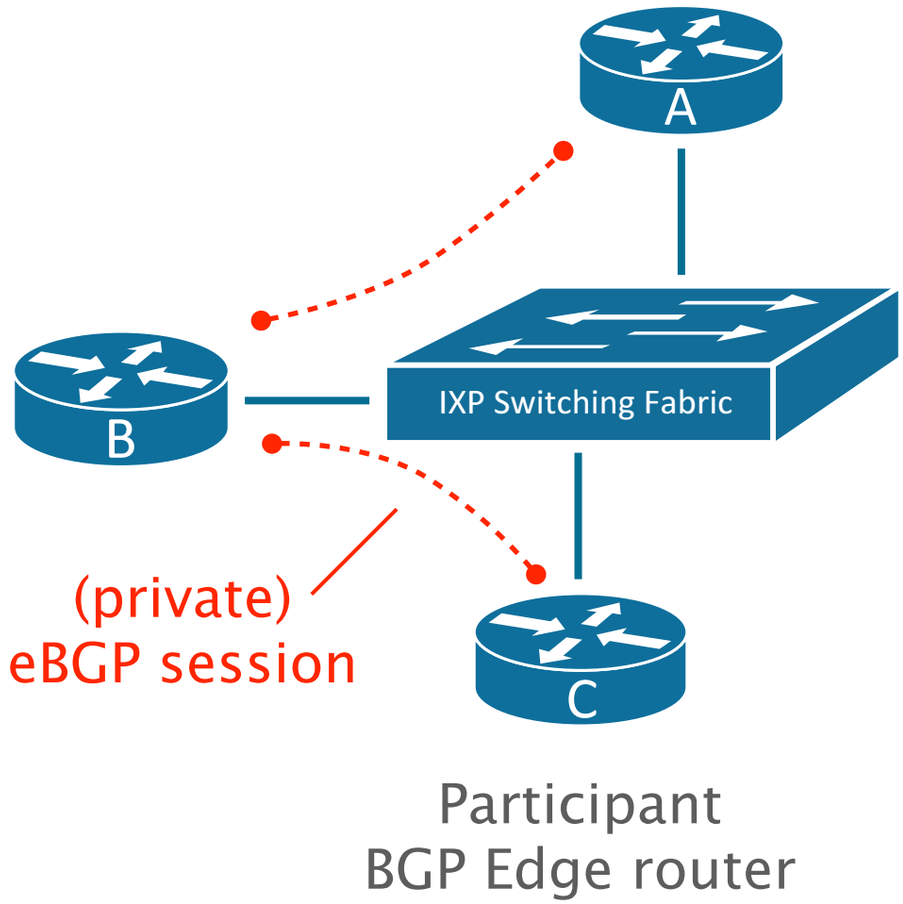
# An IXP is a large L2 domain where participants routers peer using BGP

# An IXP is a large L2 domain where participants routers peer using BGP

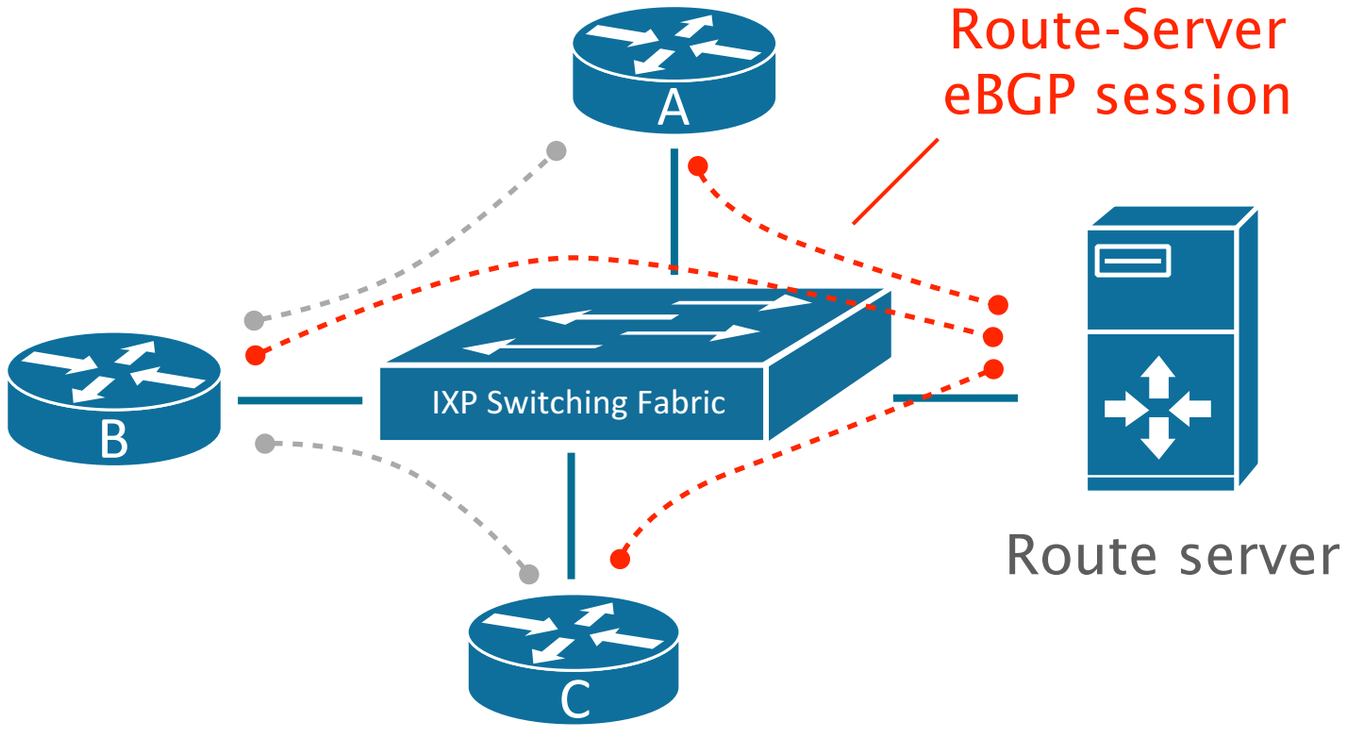



Participant
BGP Edge router

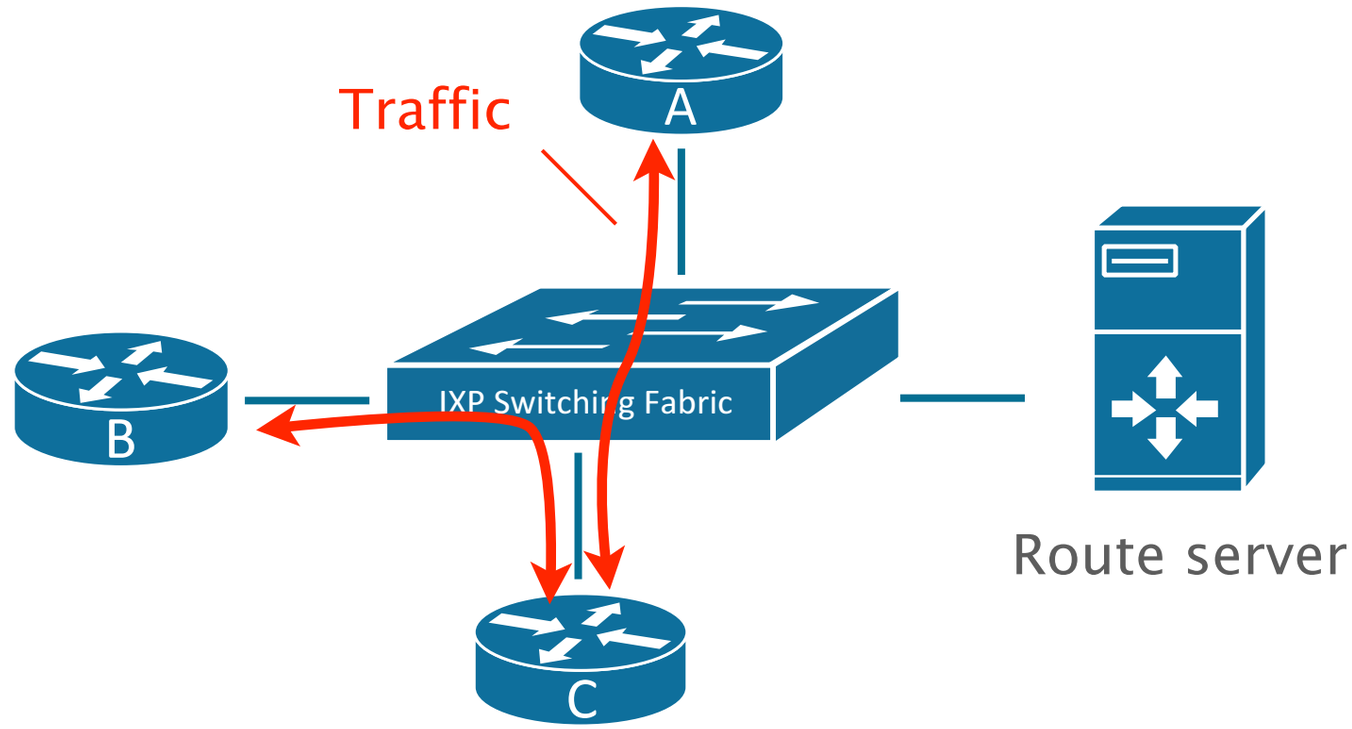
# An IXP is a large L2 domain where participants routers peer using BGP



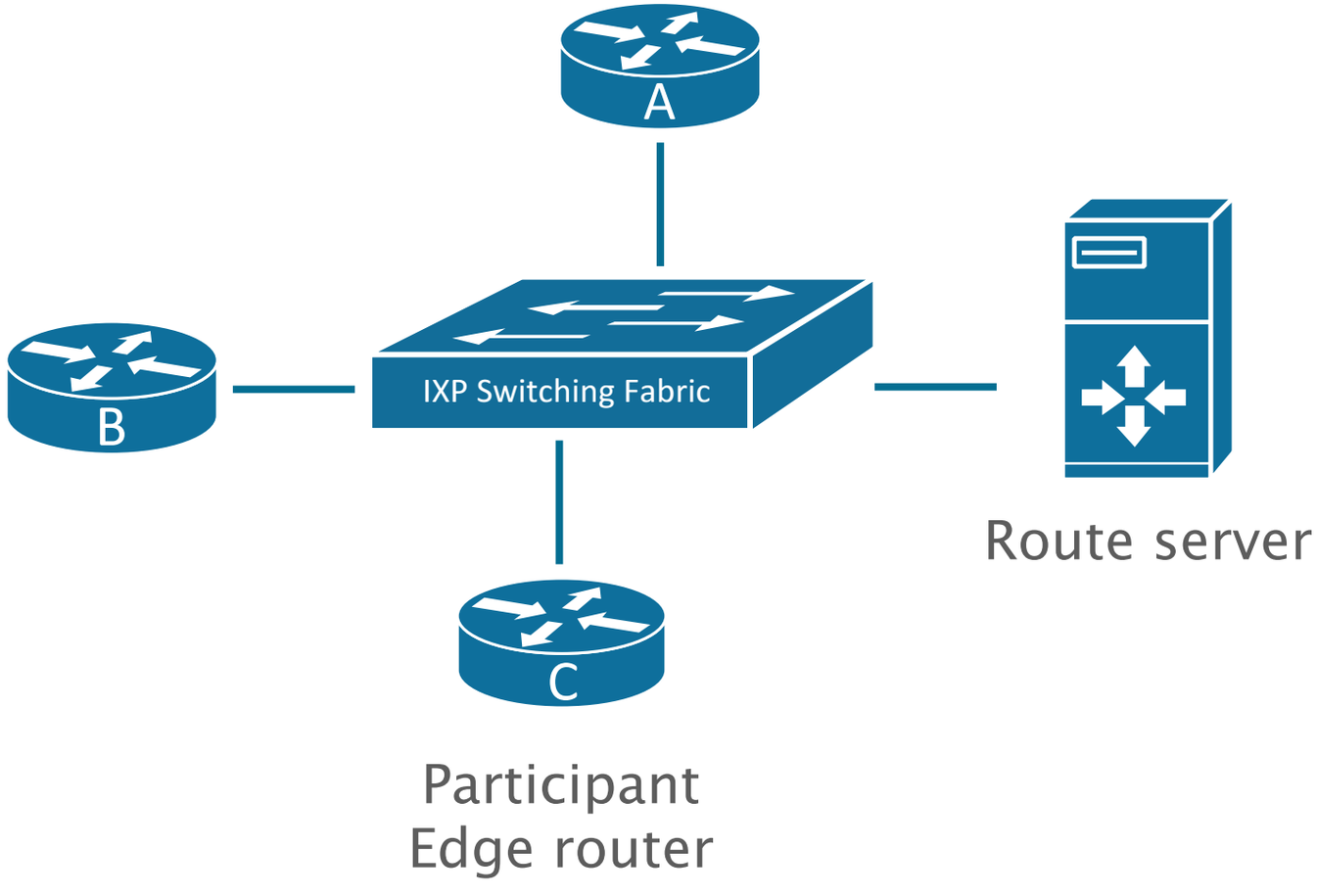
# An IXP is a large L2 domain where participants routers peer using BGP



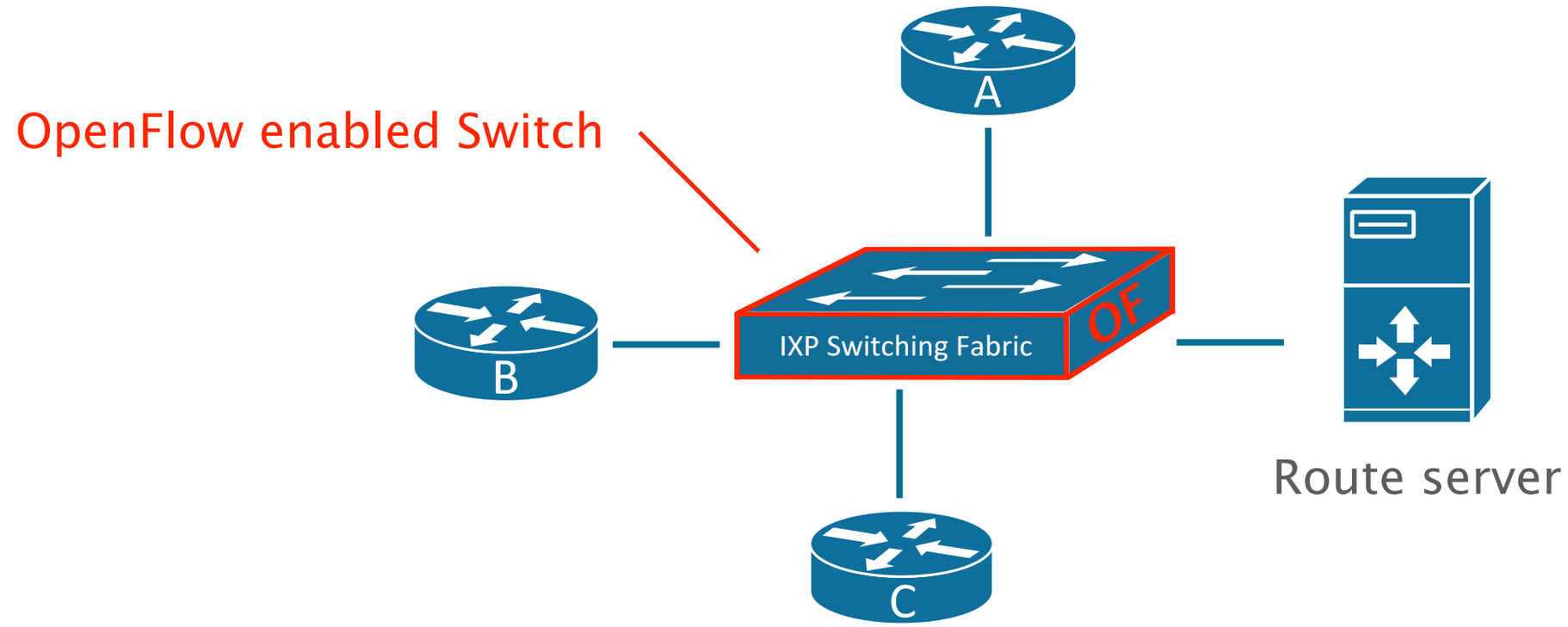
# An IXP is a large L2 domain where participants routers peer using BGP



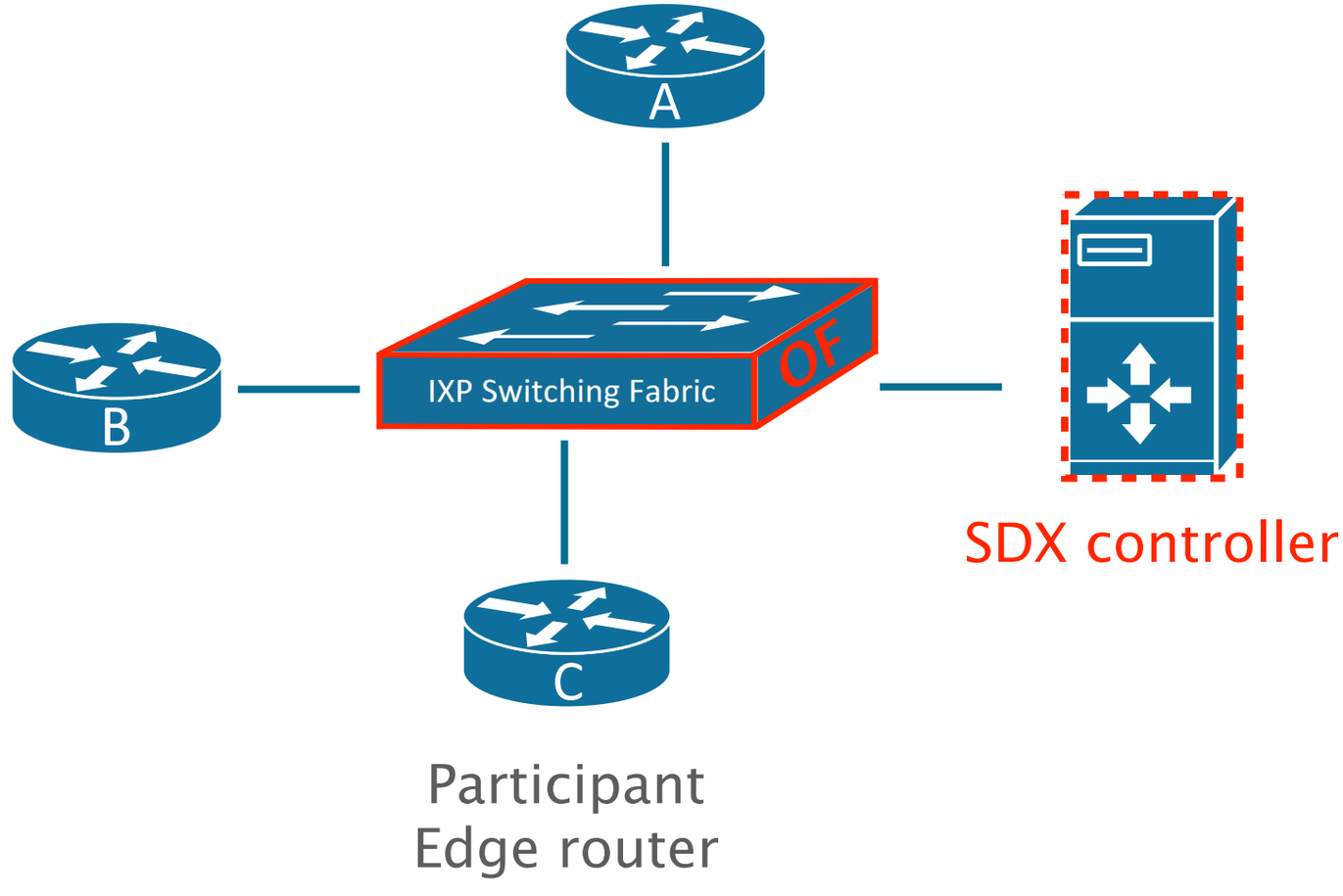
# With respect to IXPs, SDN-enabled IXPs (SDX) ...



# With respect to IXPs, SDN-enabled IXPs (SDX) *data plane* relies on SDN-capable devices



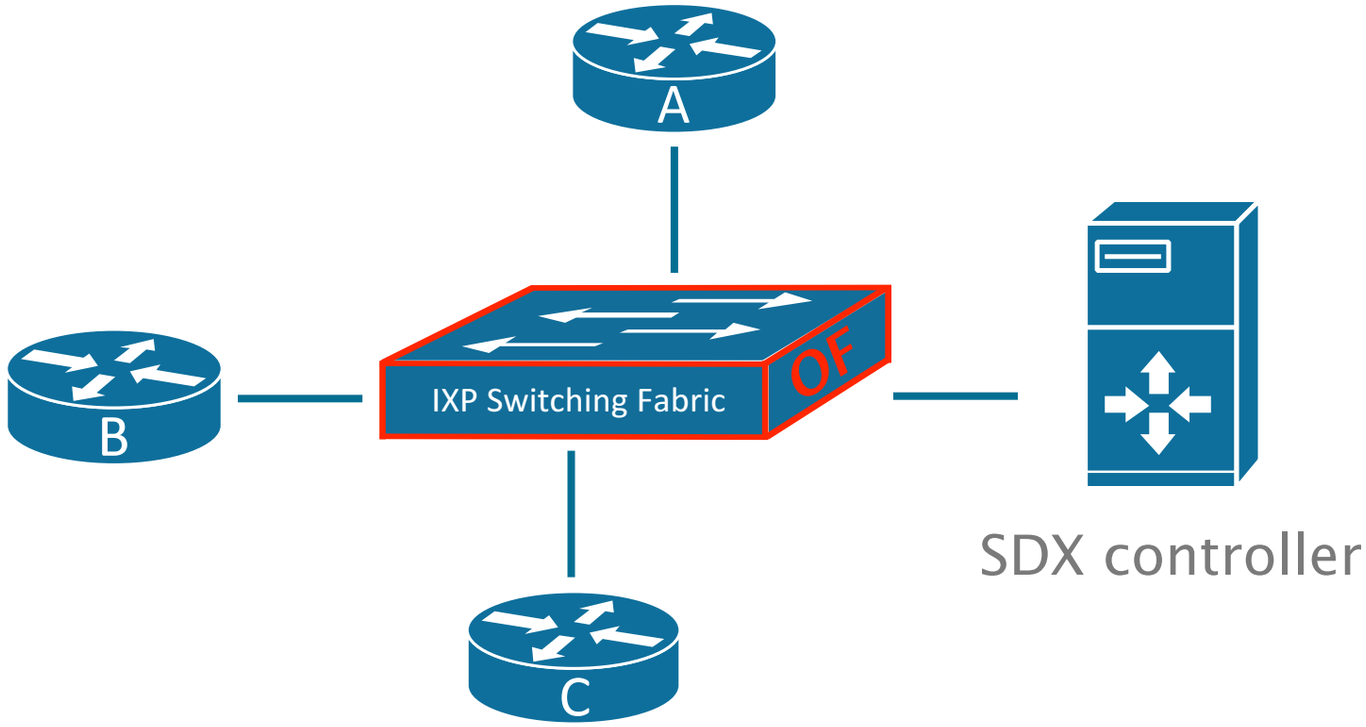
With respect to IXPs, SDN-enabled IXPs (SDX) *control plane* relies on a SDX controller



# SDX participants write policies using a high-level language on top of a virtual topology

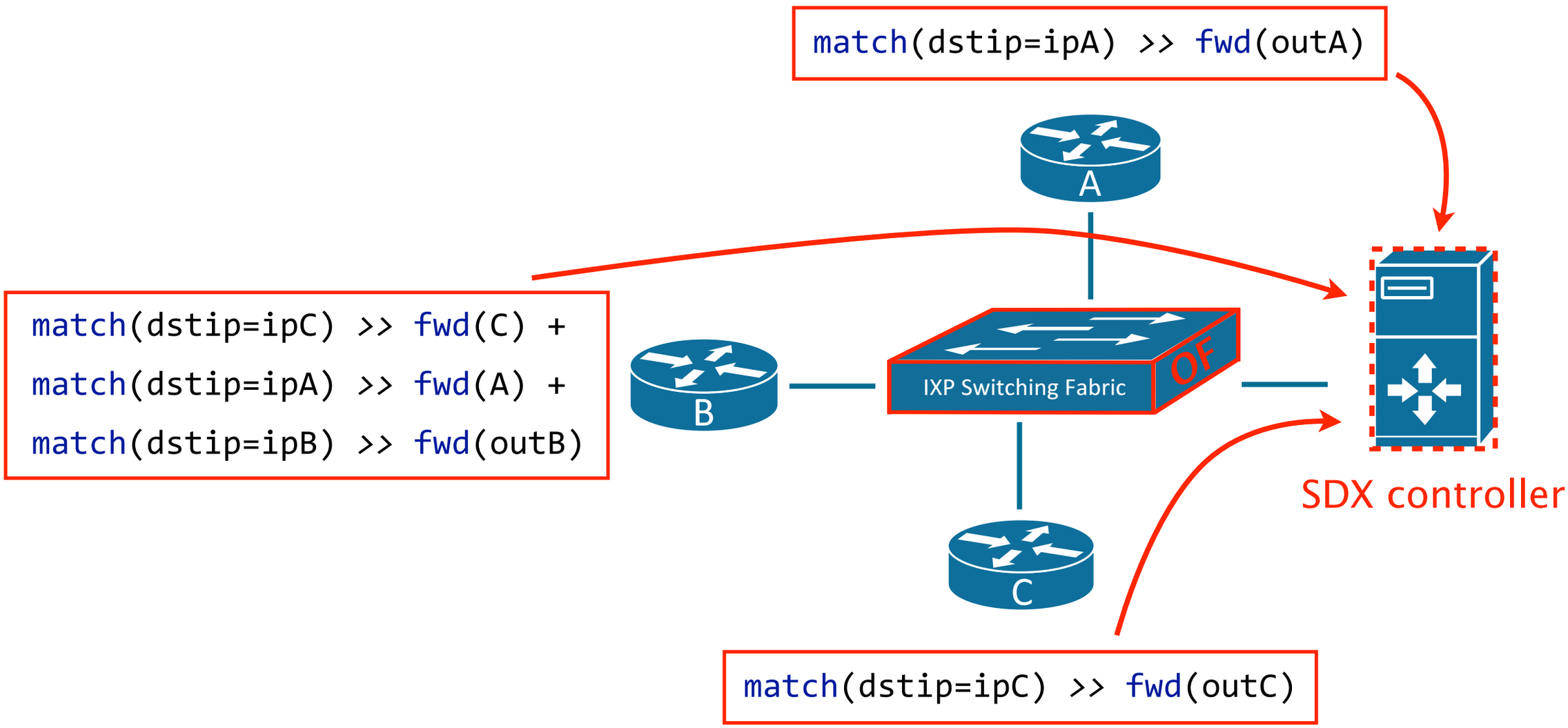
```
match(dstip=ipA) >> fwd(outA)
```

```
match(dstip=ipC) >> fwd(C) +
match(dstip=ipA) >> fwd(A) +
match(dstip=ipB) >> fwd(outB)
```

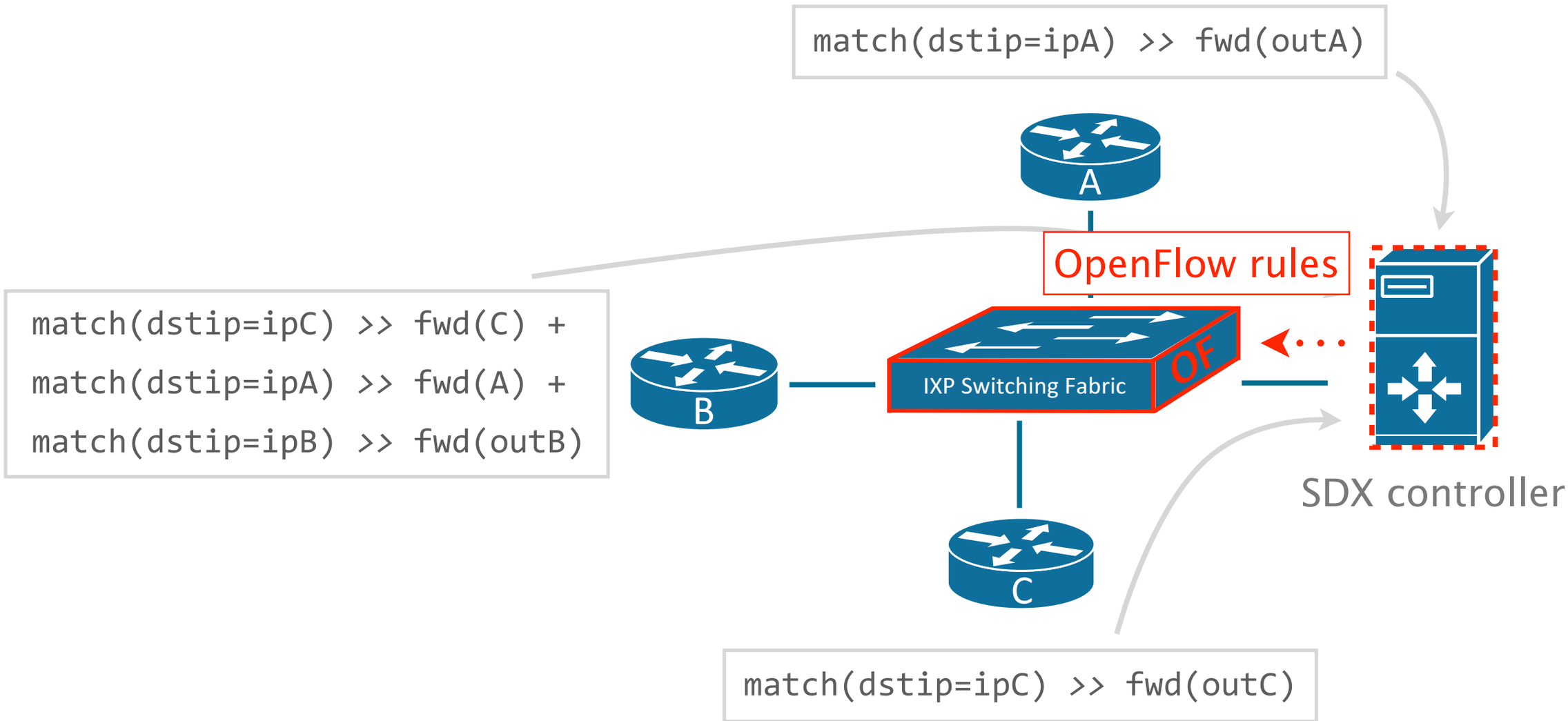



```
match(dstip=ipC) >> fwd(outC)
```

# The SDX controller composes policies together ensuring *isolation* and *correctness*

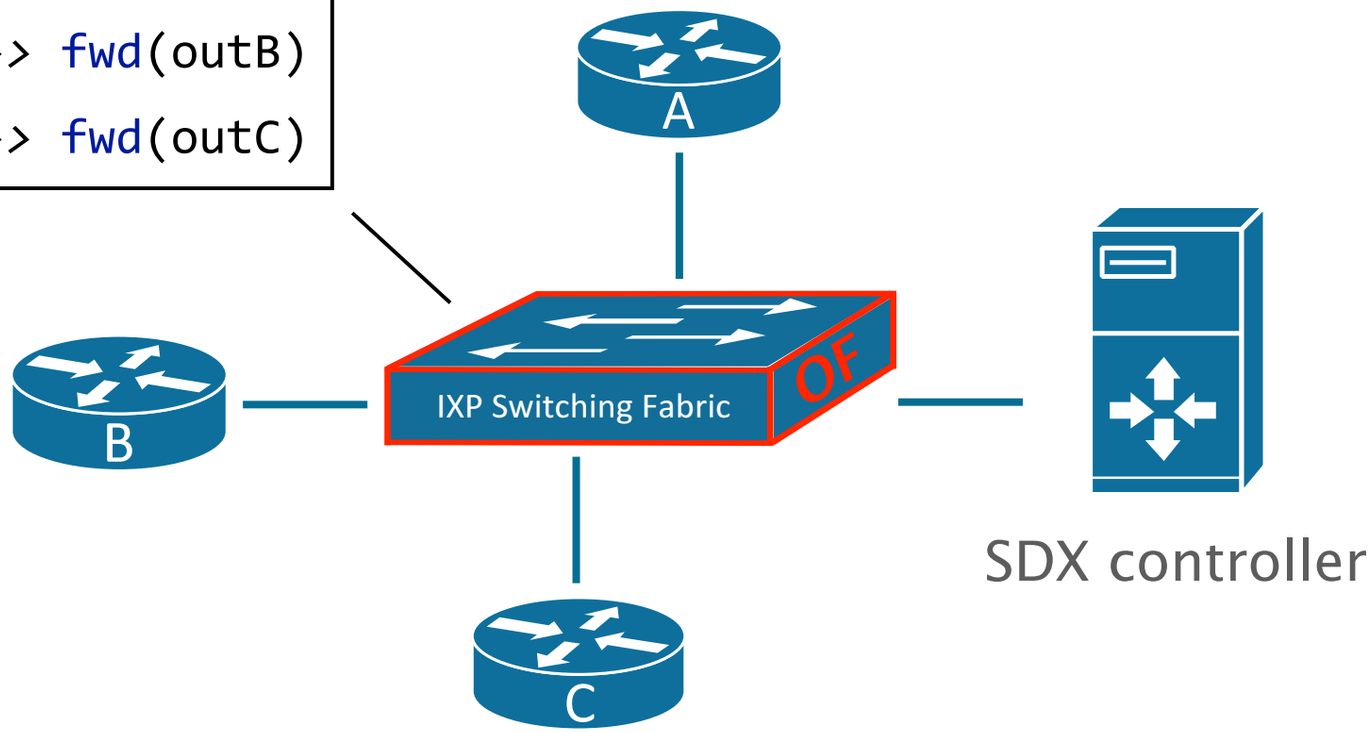


# The SDX controller composes policies together ensuring *isolation* and *correctness*

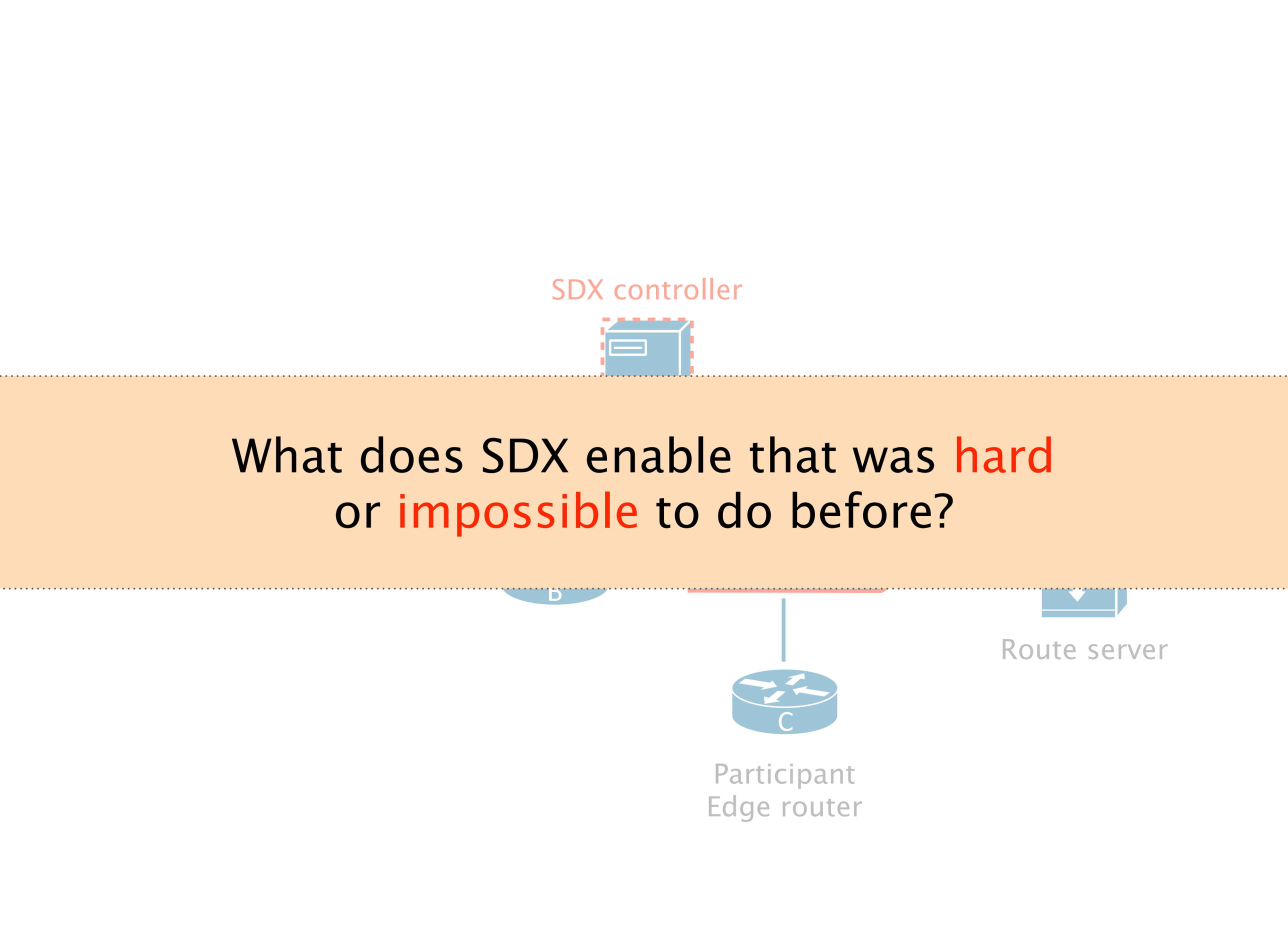


# To ensure compatibility and scalability, SDX supports MAC-based forwarding by default

```
match(dstmac=MACA) >> fwd(outA)
match(dstmac=MACB) >> fwd(outB)
match(dstmac=MACC) >> fwd(outC)
```




*Participants' policies subsumes default forwarding behavior*




# What does SDX enable that was hard or impossible to do before?

# SDX enables a wide range of novel interdomain applications

| | |
|---|---|
| **security** | Prevent/block policy violation |
| | Prevent participants communication |
| **forwarding optimization** | Middlebox traffic steering |
| | Traffic offloading |
| | Inbound Traffic Engineering |
| **peering** | Application-specific peering |
| **remote-control** | Wide-area load balancing |
| | Influence BGP path selection |
| | Upstream blocking of DoS attacks |

# SDX enables a wide range of novel interdomain applications

| | |
|---|---|
| security | Prevent/block policy violation |
| | Prevent participants communication |
| **forwarding optimization** | Middlebox traffic steering |
| | Traffic offloading |
| | **Inbound Traffic Engineering** |
| peering | Application-specific peering |
| **remote-control** | **Wide-area load balancing** |
| | Influence BGP path selection |
| | **Upstream blocking of DoS attacks** |

# Novel Applications for a SDN-enabled Internet Exchange Point

1 Inbound Traffic Engineering

2 Upstream DoS blocking

3 Wide-area load balancing

# Novel Applications for a SDN-enabled Internet Exchange Point

1 Inbound Traffic Engineering

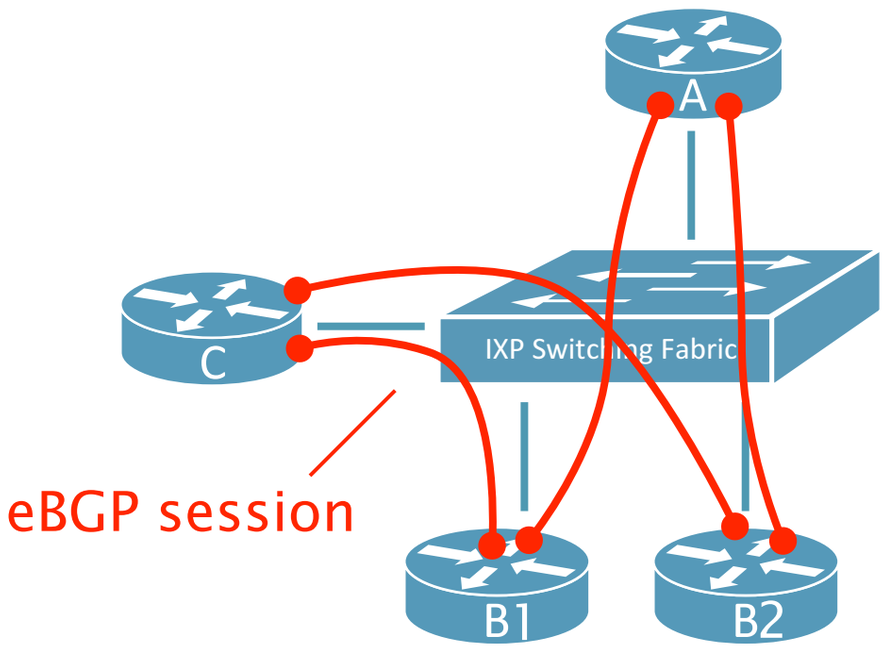
Upstream DoS blocking

Wide-area load balancing

# SDX can improve inbound traffic engineering

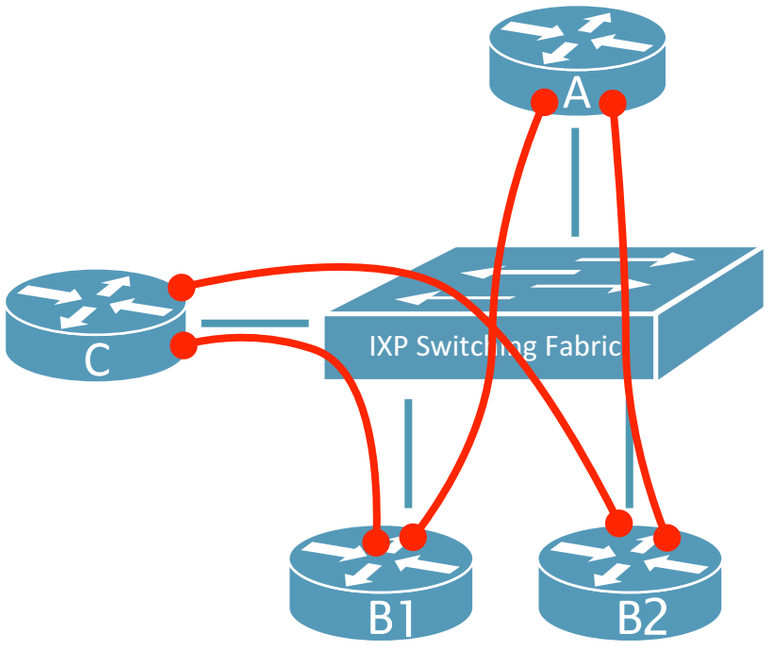
# SDX can improve inbound traffic engineering

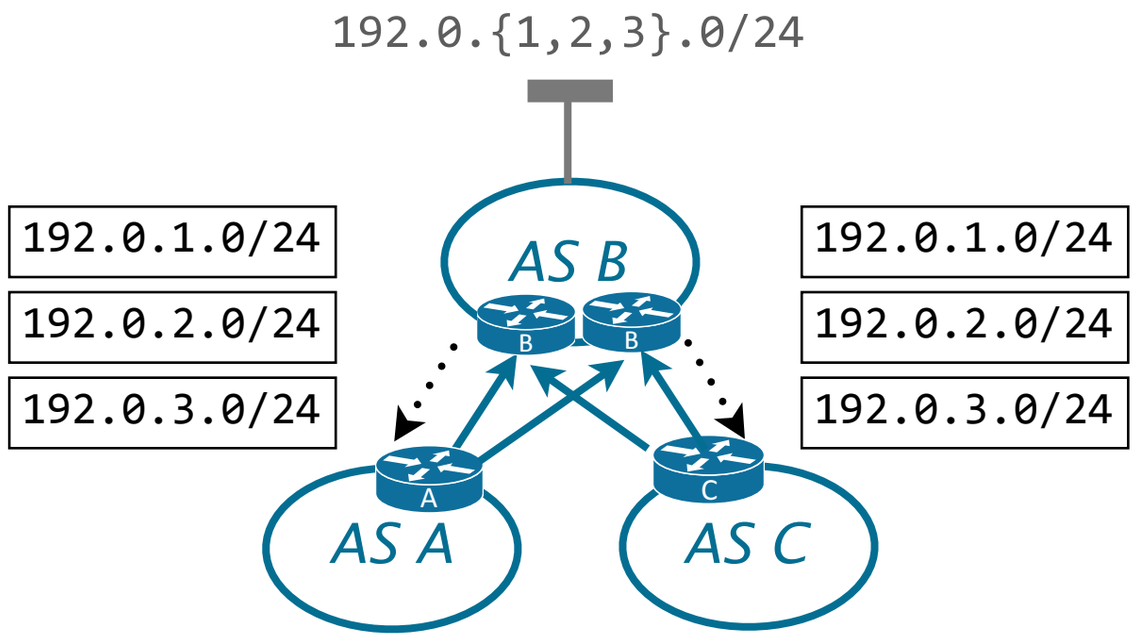
Given an IXP Physical Topology



# SDX can improve inbound traffic engineering

Given an IXP Physical Topology and a BGP topology

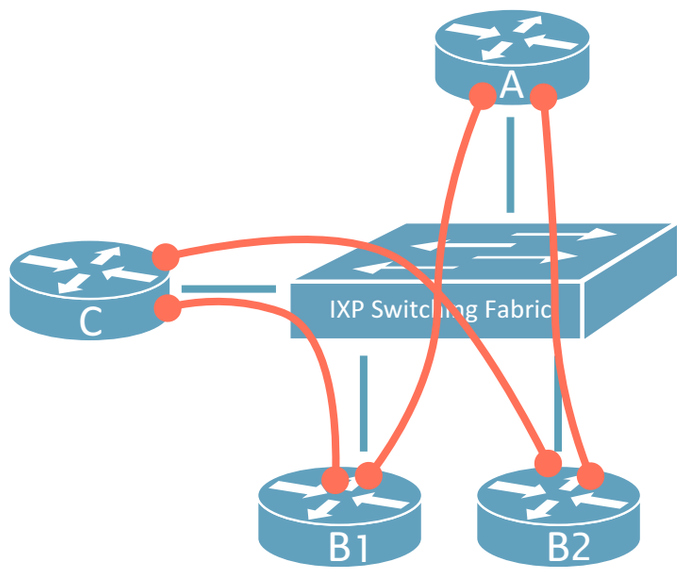


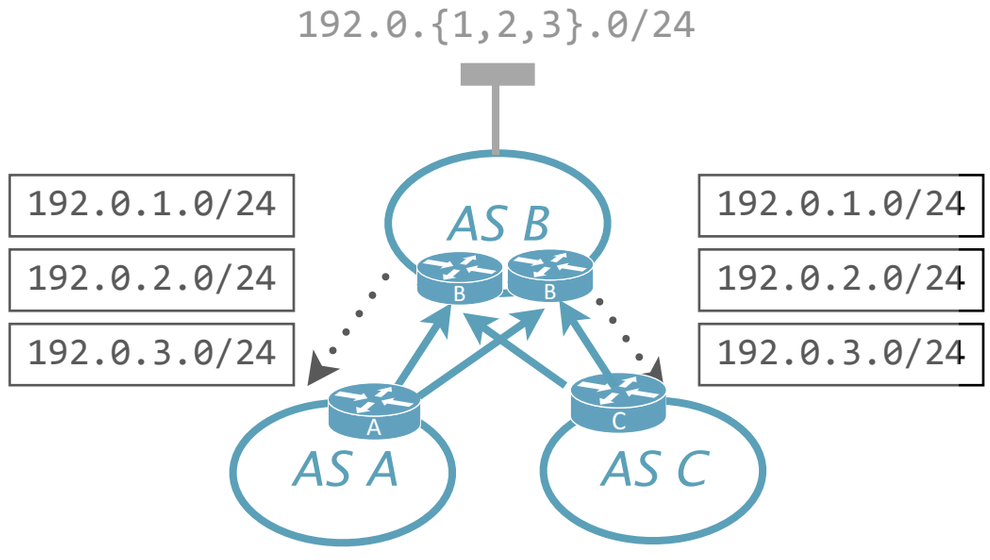
# SDX can improve inbound traffic engineering

## Implements B's inbound policy

| to | from | receive on |
|---|---|---|
| 192.0.1.0/24 | A | B1 |
| 192.0.2.0/24 | B | B2 |
| 192.0.2.0/24 | ATT_IP | B2 |
| 192.0.2.0/24 | * | B1 |
| 192.0.3.0/24 | * | B2 |




IXP Topology

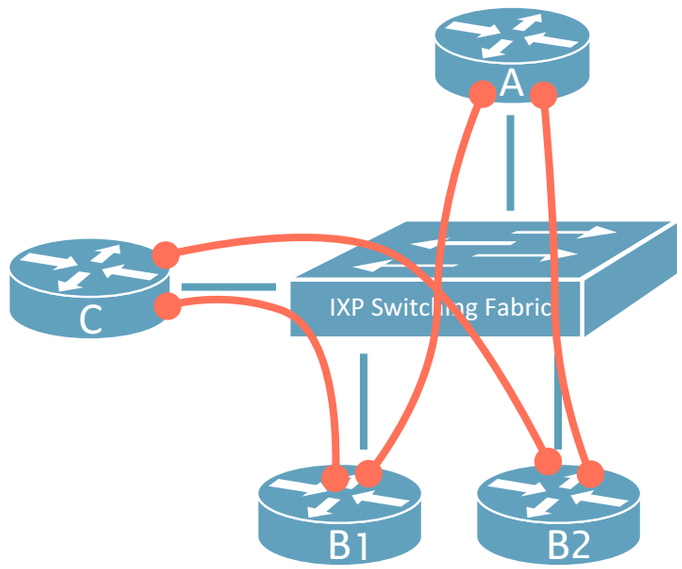



BGP Topology

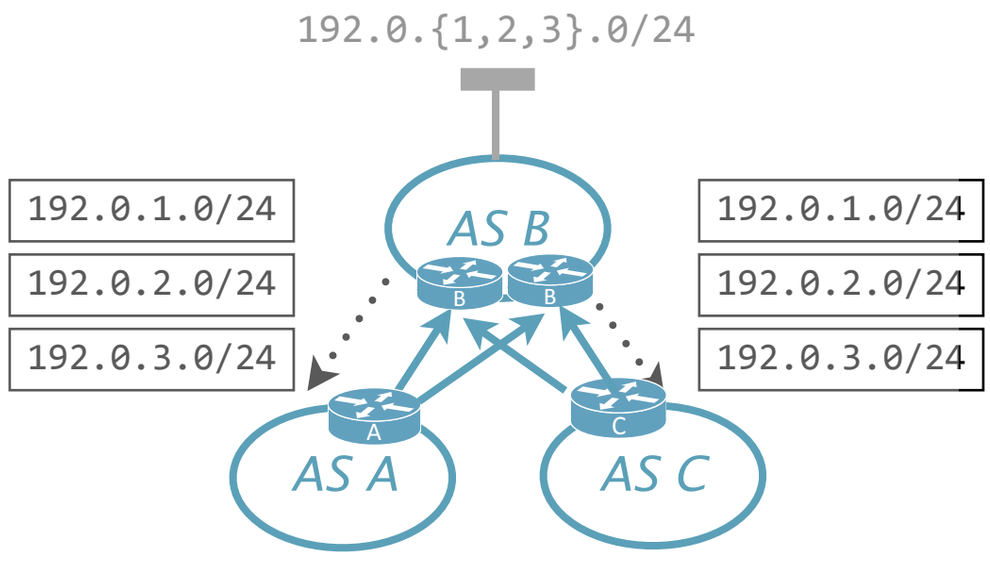
# How do you do that with BGP?

Implements B's inbound policy

| to | from | receive on |
|---|---|---|
| 192.0.1.0/24 | A | B1 |
| 192.0.2.0/24 | B | B2 |
| 192.0.2.0/24 | ATT_IP | B2 |
| 192.0.2.0/24 | * | B1 |
| 192.0.3.0/24 | * | B2 |




IXP Topology




BGP Topology

# It is at least hard... BGP provides few knobs to influence remote decisions

Implementing such a policy is configuration-intensive

using AS-Path prepend, MED, community tagging, etc.

and even impossible for some requirements...

BGP policies **cannot** influence remote parties
decisions based on source addresses

| to | **from** | receive on |
|---|---|---|
| `192.0.2.0/24` | **`ATT_IP`** | B2 |

# In any case, the outcome is unpredictable

Implementing such a policy is configuration-intensive

using AS-Path prepend, MED, community tagging, etc.

Absolutely no guarantee that the remote party will comply

one can only “influence” remote decisions

Networks engineers have no choice but to “try and see”

which makes it impossible to adapt to traffic pattern

# With a SDX, implementing B's inbound policy is easy

SDX policies give B *direct* control on its forwarding paths

| to | from | fwd |
|---|---|---|
| 192.0.1.0/24 | A | B1 |
| 192.0.2.0/24 | B | B2 |
| 192.0.2.0/24 | ATT_IP | B2 |
| 192.0.2.0/24 | * | B1 |
| 192.0.3.0/24 | * | B2 |

B's SDX Policy

```
match(dstip=192.0.1.0/24, srcmac=A) >> fwd(B1)

match(dstip=192.0.2.0/24, srcmac=B) >> fwd(B2)

match(dstip=192.0.2.0/24, srcip=ATT) >> fwd(B2)

match(dstip=192.0.2.0/24) >> fwd(B1)

match(dstip=192.0.3.0/24) >> fwd(B2)
```

# Novel Applications for a SDN-enabled Internet Exchange Point

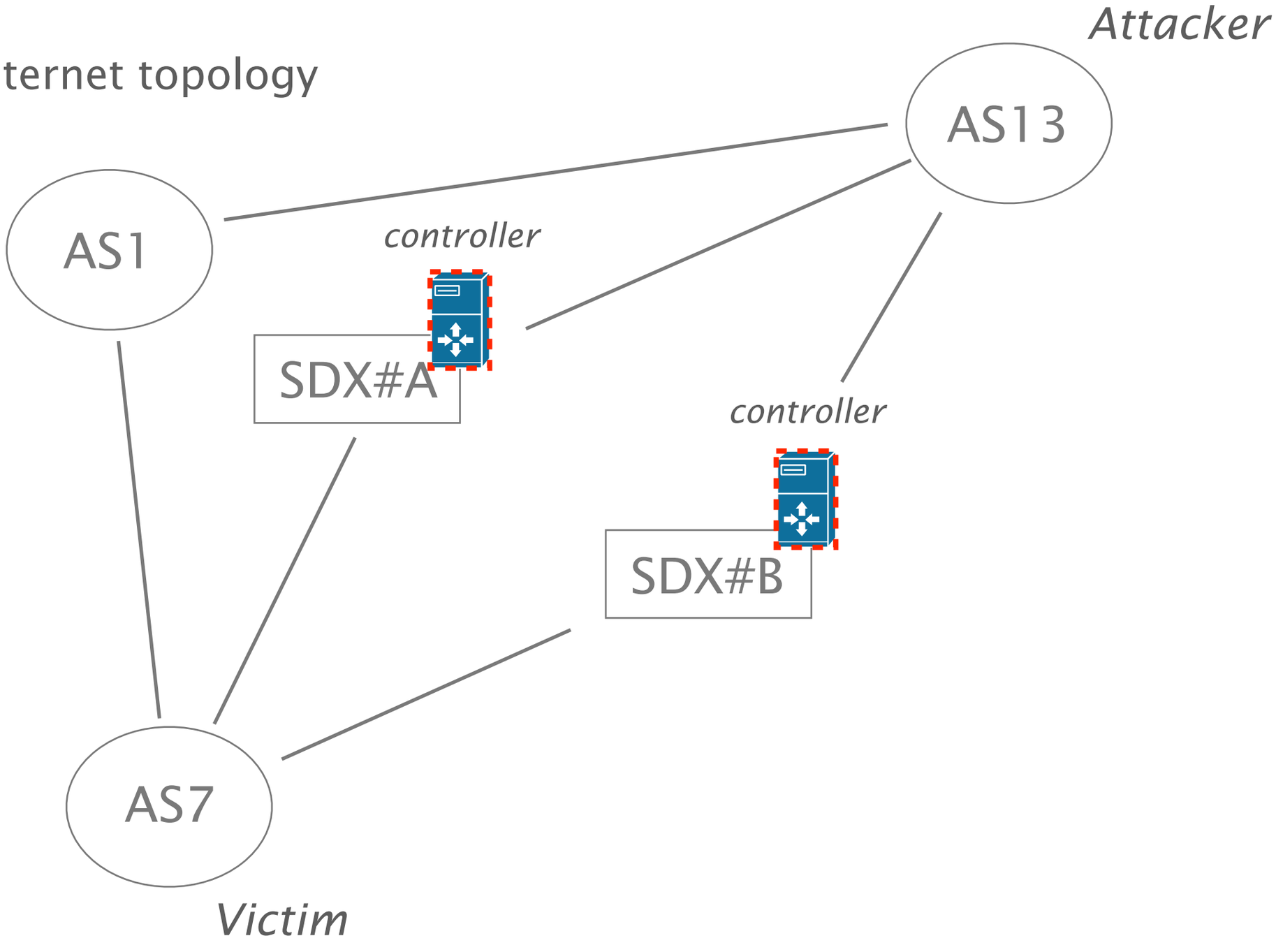
Inbound Traffic Engineering

2 Upstream DoS blocking

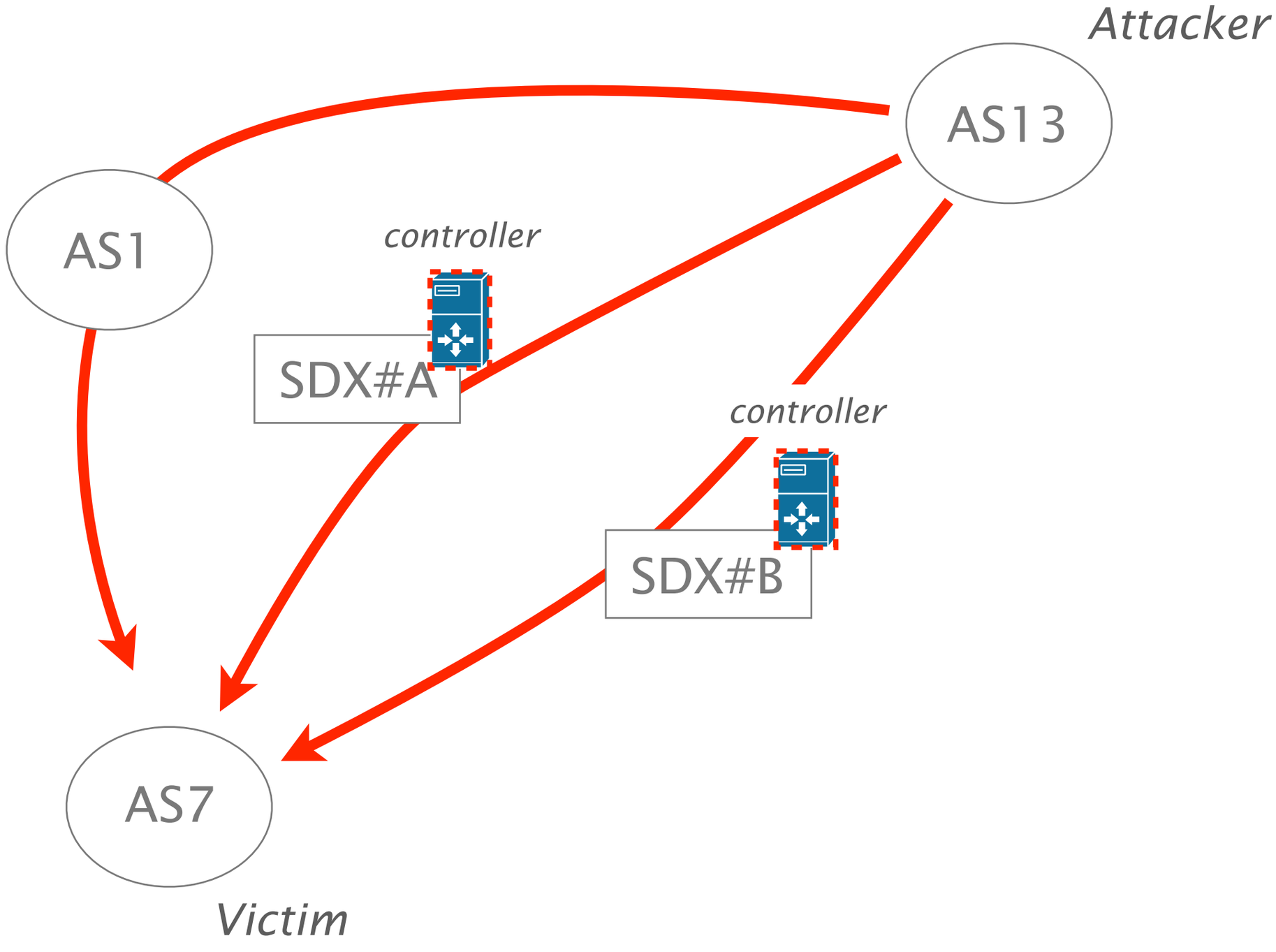
Wide-area load balancing

# SDX can help blocking DoS attacks closer to the source

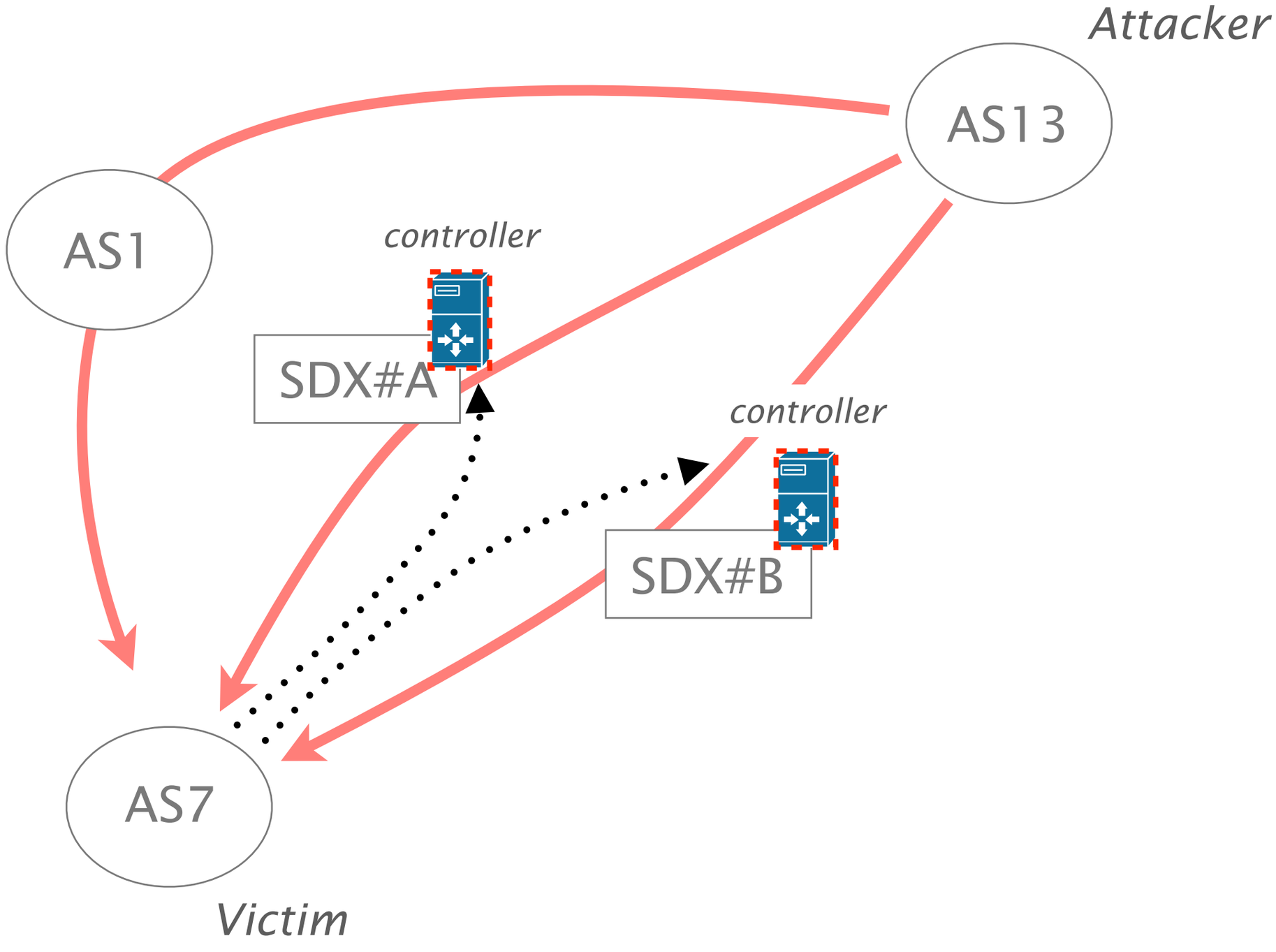
A simple Internet topology



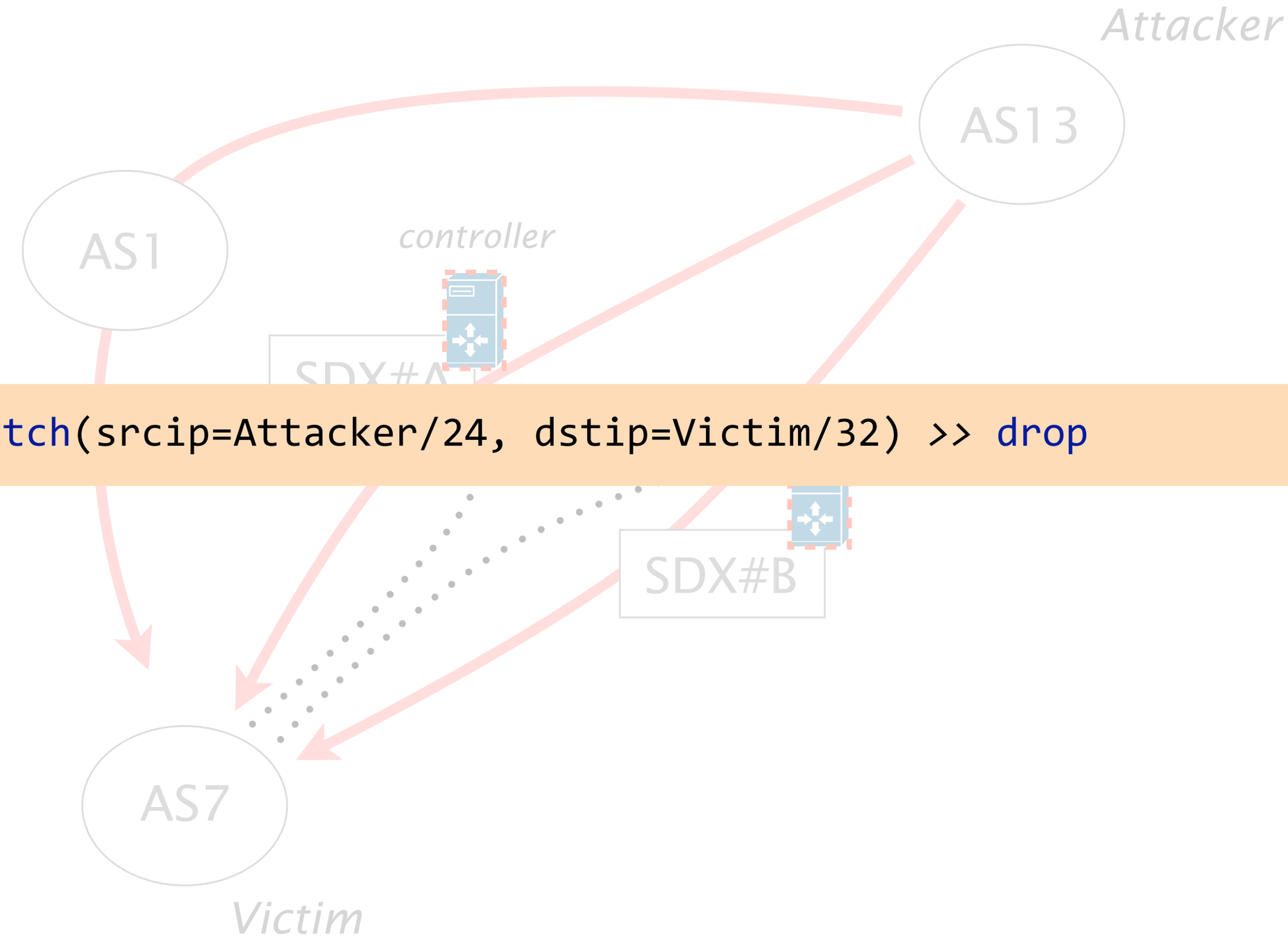
# AS7 is under a DoS attack originated by AS13



# AS7 can remotely install drop rule in the SDX platforms



# AS7 can remotely install drop rule in the SDX platforms




```
match(srcip=Attacker/24, dstip=Victim/32) >> drop
```

# SDX-based DoS protection is more efficient than traditional blackholing solutions

Remote ASes could drop traffic destined to their network

even if there are not physically connect to the IXP!

Traffic drop can be done based on any field

source address, destination address, port number, etc.

Traffic drop can be coordinated across multiple IXPs

thanks to SDX controllers collaboration

# Novel Applications for a SDN-enabled Internet Exchange Point

Inbound Traffic Engineering

Application-specific peering

3 Wide-area load balancing

# DNS-based wide-area load balancing has several limitations

High TTL values lead to slow recovery when a server fails

due to caching by local DNS servers and browsers

Low TTL values lead to higher delay for DNS resolution

due to cache misses

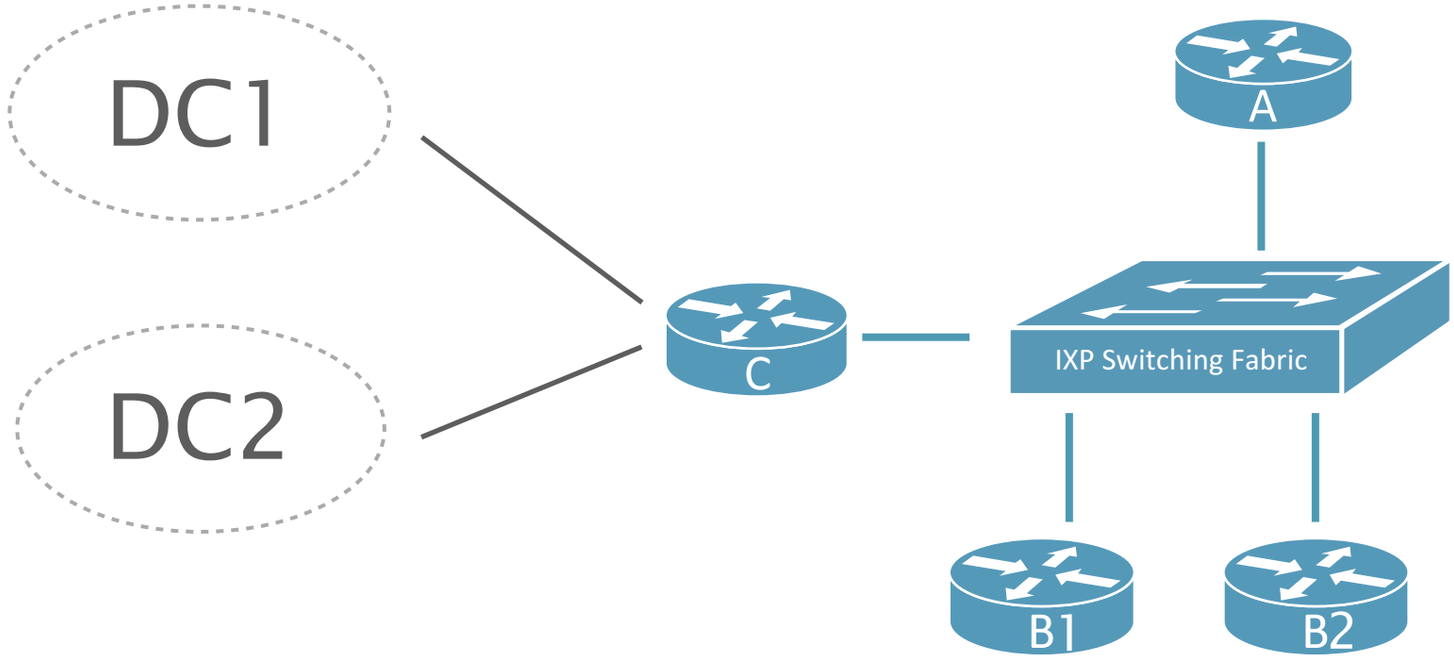
Load-balancing is not based on the client IP address

but on the IP address of the DNS resolver (e.g., Google DNS)

# SDX enable direct and quick control of traffic redirection

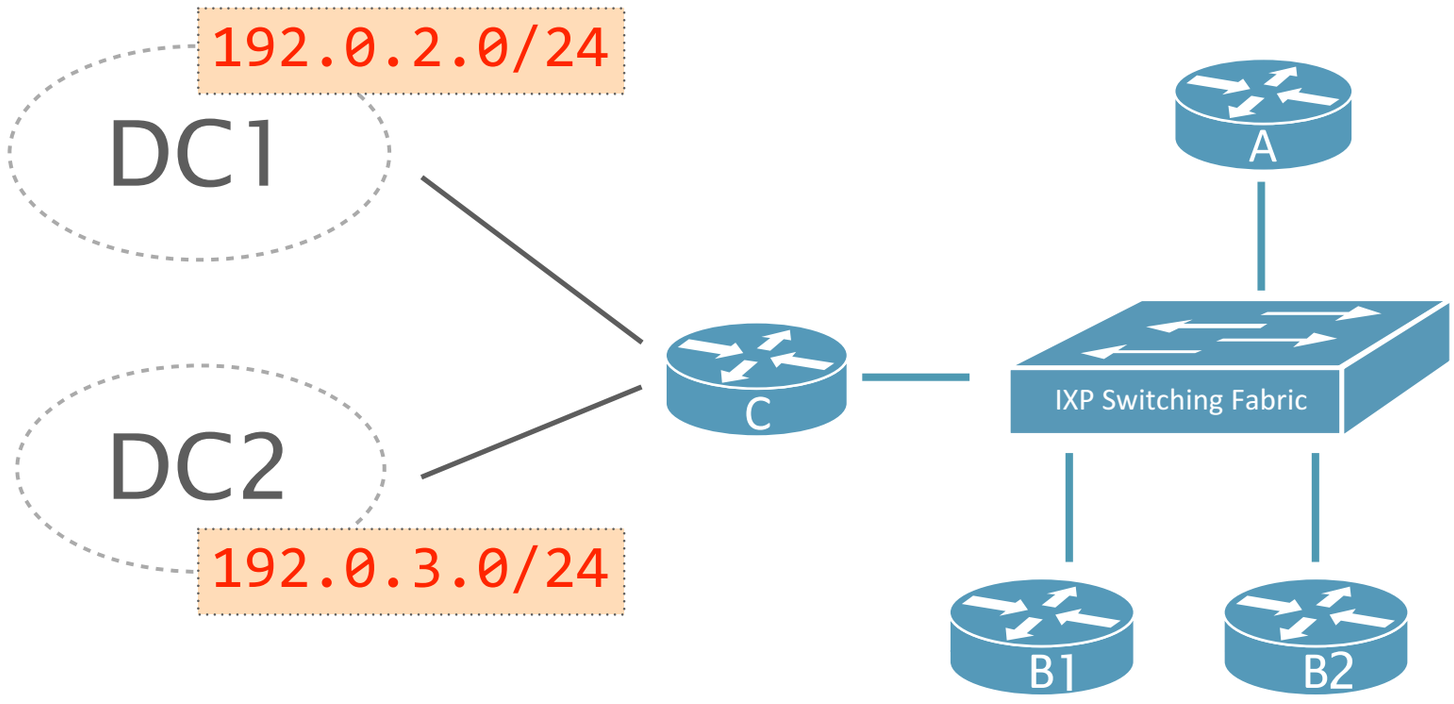
# SDX enable direct and quick control of traffic redirection

C is a CDN hosting three types of services:



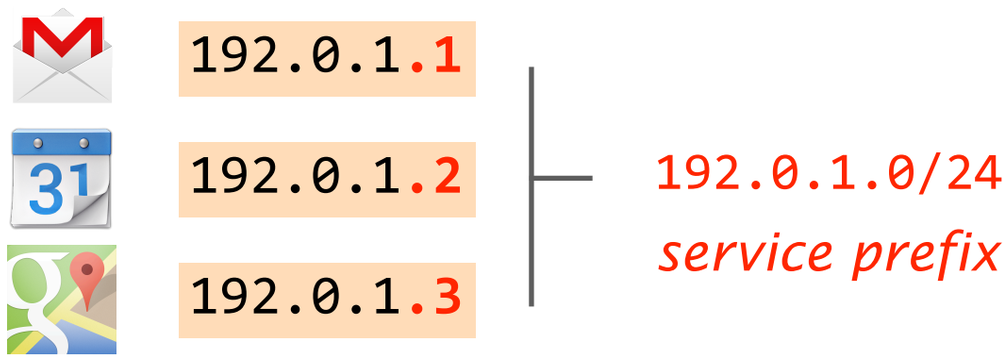
# SDX enable direct and quick control of traffic redirection

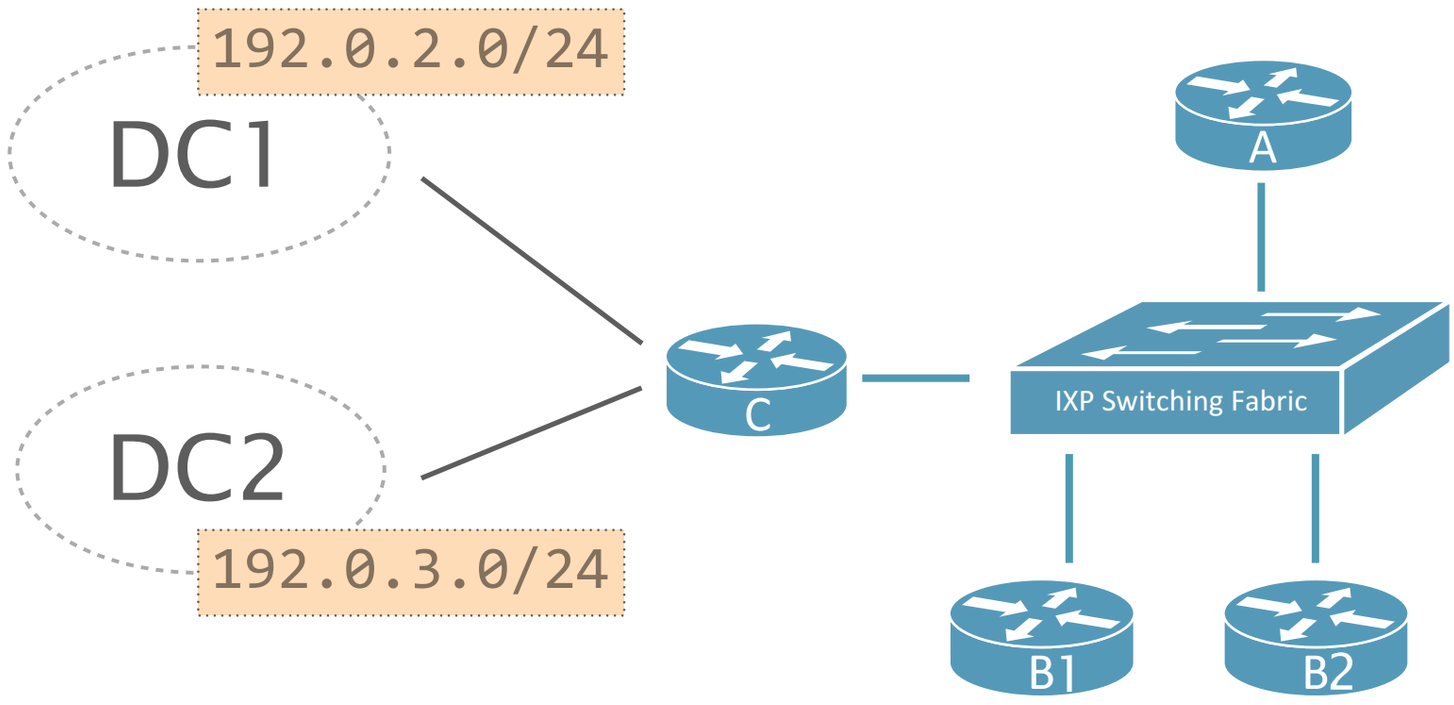
Each of C's data center is assigned to a single IP prefix



# SDX enable direct and quick control of traffic redirection

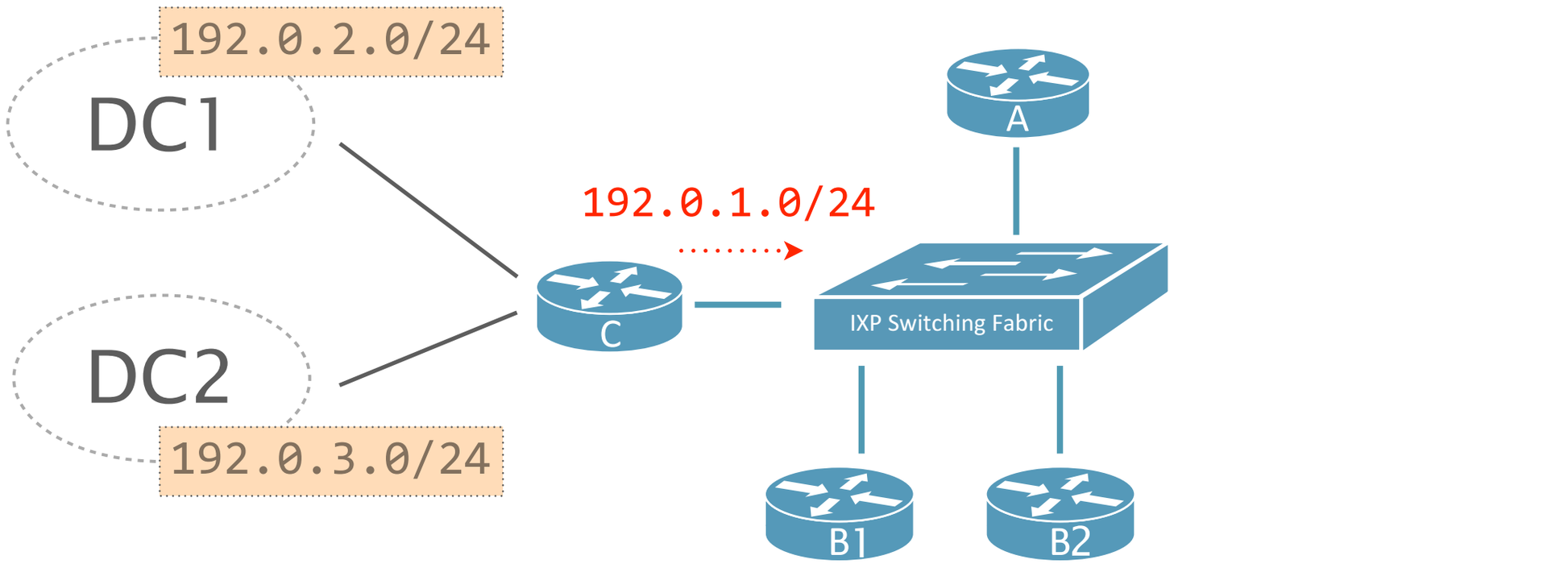
C assigns one IP address per service taken from a *service prefix*



# SDX enable direct and quick control of traffic redirection

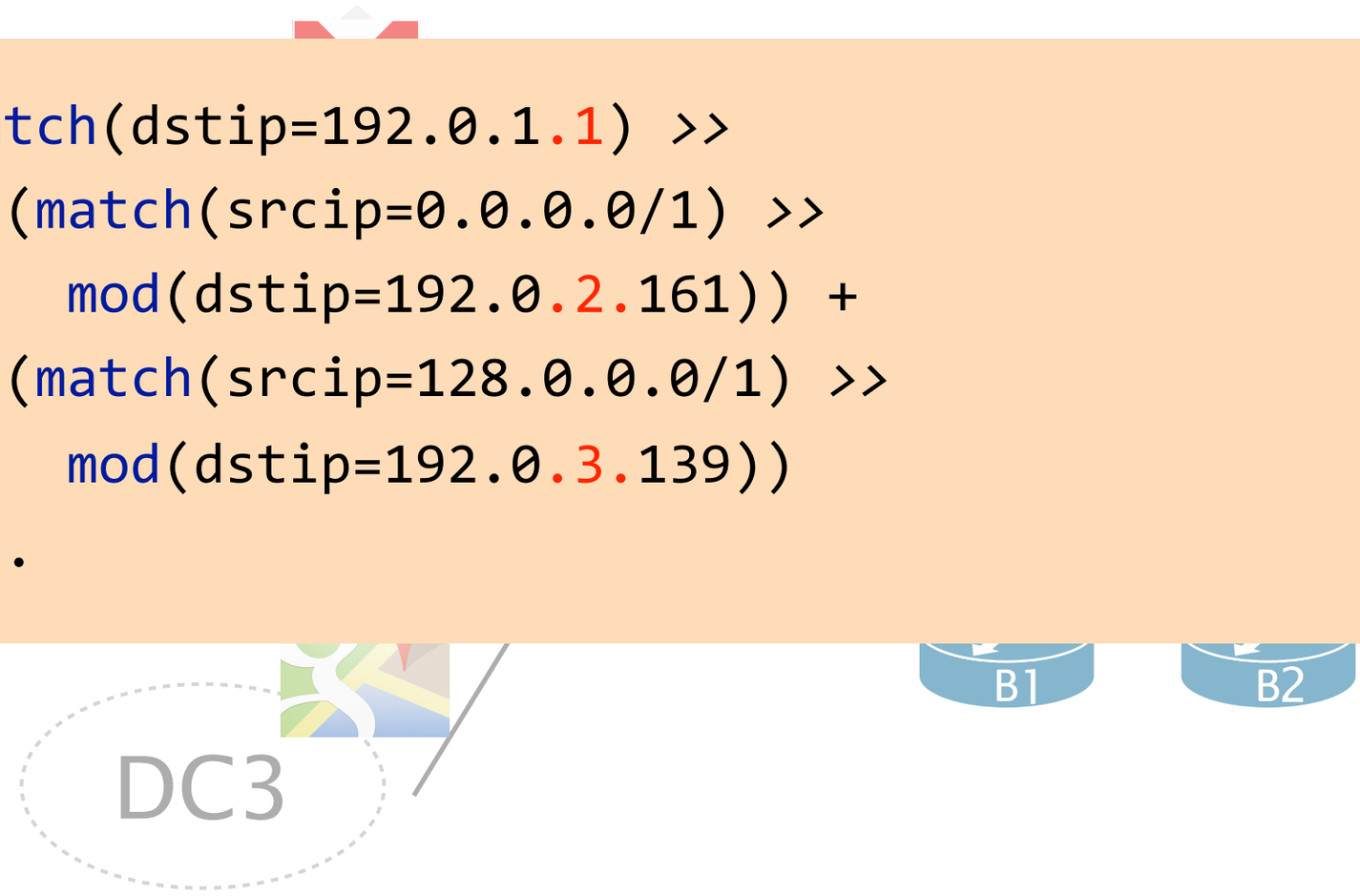
C announces the service prefix directly from the IXP



# SDX enable direct and quick control of traffic redirection

C installs a policy to direct requests to a given service to the appropriate replica based on the client's IP address

```
match(dstip=192.0.1.1) >>
  (match(srcip=0.0.0.0/1) >>
    mod(dstip=192.0.2.161)) +
  (match(srcip=128.0.0.0/1) >>
    mod(dstip=192.0.3.139))
...
```



# SDX enable direct and quick control of traffic redirection

SDX load-balancing is

- fast — no more problem due to DNS caching
- flexible — any load-balancing algorithm can be used
- efficient — based on the actual client IP address

# Novel Applications for a SDN-enabled Internet Exchange Point

Inbound Traffic Engineering

Application-specific peering

Wide-area load balancing

# We have running code
# as well as a first deployment site

We have implemented a first SDX controller prototype

which supports policies composition

We have partnered with a large regional IXP in Atlanta

which hosts many large content providers such as Akamai

Ping me if you are interested in knowing more

# Several challenges remain

We need authentication mechanisms to validate policies

*e.g.*, using RPKI

We need “access-lists” to constrain the policies

*e.g.*, limiting the capabilities available to each participant

We need to make the platform scalable

as SDN devices currently support a relatively small # of rules

# Novel Applications for a SDN-enabled Internet Exchange Point

Laurent Vanbever

Laurent Vanbever

http://vanbever.eu

SDN Research Group, IETF87

July, 29 2013