# Edge Fabric:
## Delivering Oceans of Content to the World

**Brandon Schlinker**[1,2]

Hyojeong Kim[1], Timothy Cui[1], Ethan Katz-Bassett[2,3], Harsha V. Madhyastha[4], Italo Cunha[5]
James Quinn[1], Saif Hasan[1], Petr Lapukhov[1], James Hongyi Zeng[1]

[1] Facebook, [2] University of Southern California, [3] Columbia University,
[4] University of Michigan, [5] Universidade Federal de Minas Gerais



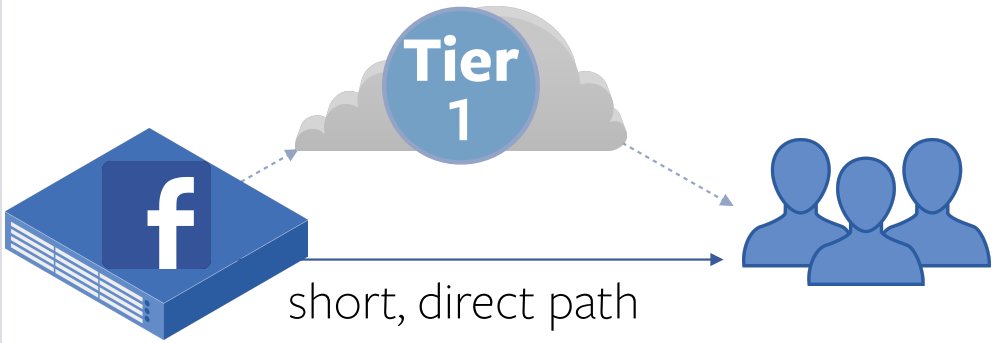
IETF 104, March 2019

# **Facebook's** Global Network

points of presence
**around the world**

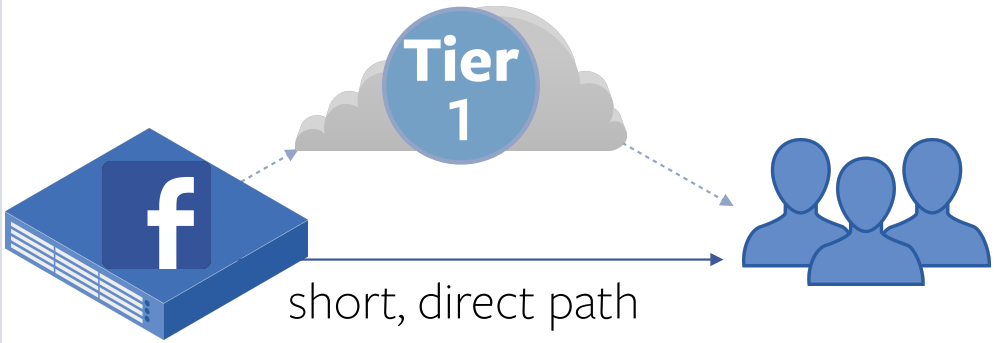
interconnect with
**thousands of networks**

# Benefits of Rich Interconnection

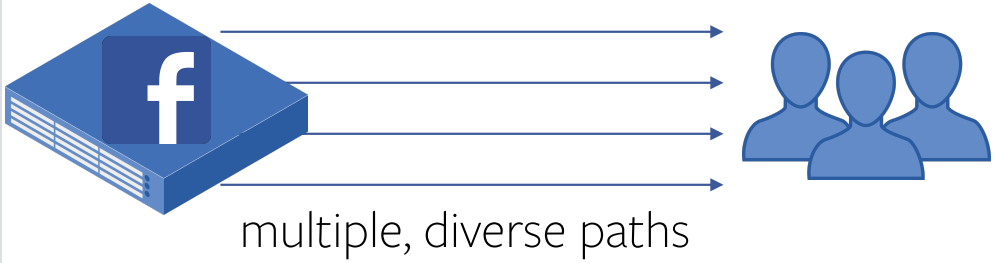



short, direct paths
bypass transit providers

# Benefits of Rich Interconnection




short, direct paths
bypass transit providers



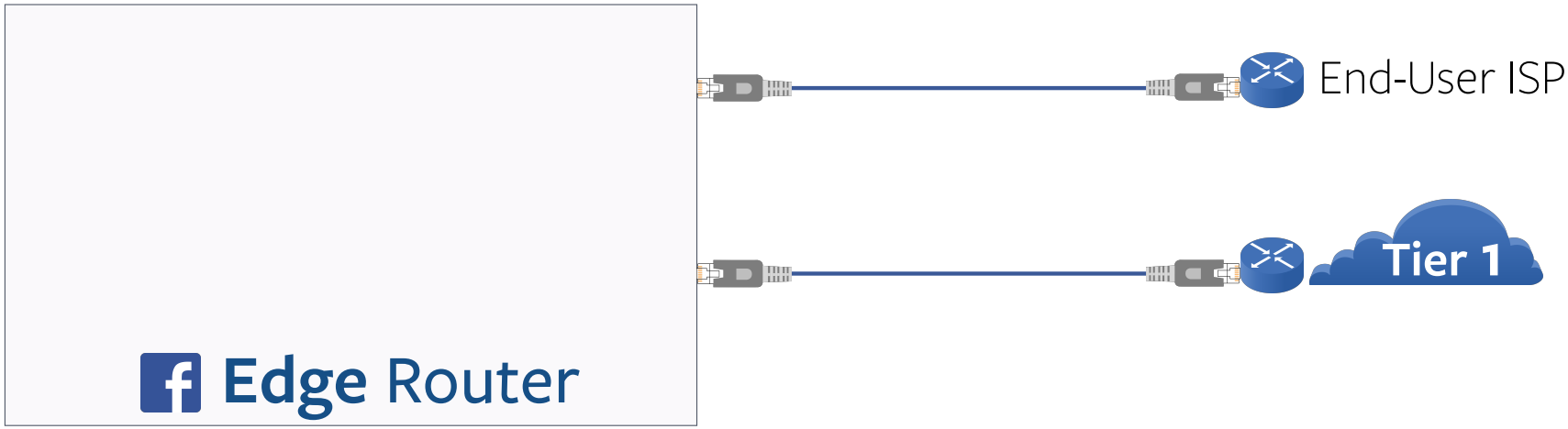

substantial path diversity

# Basics of Interconnection at a PoP

# Basics of Interconnection at a PoP

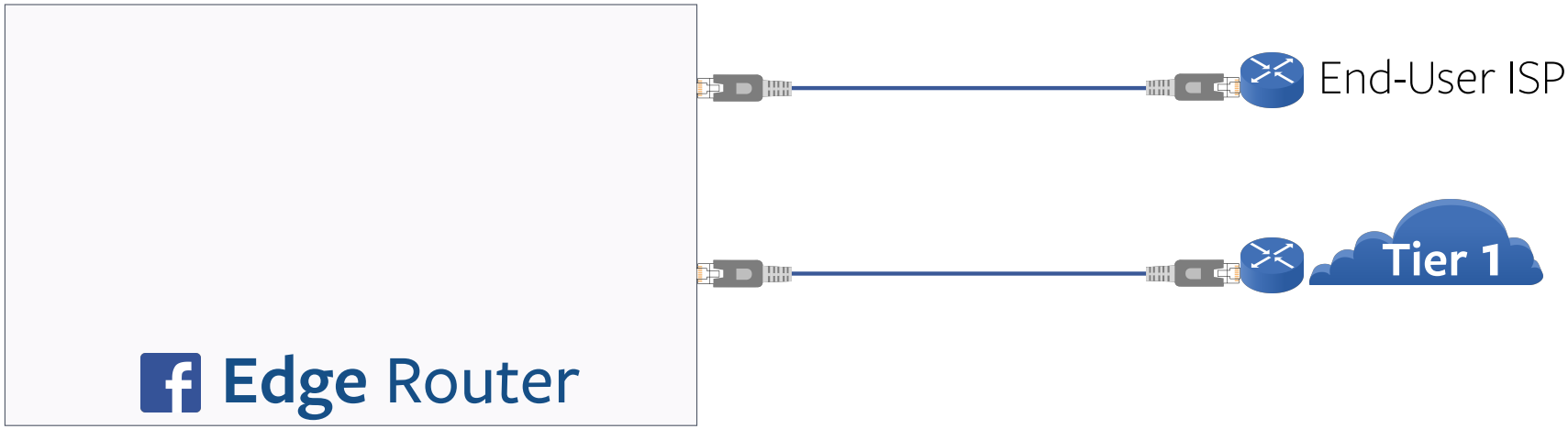
# Basics of Interconnection at a PoP

# Basics of Interconnection at a PoP



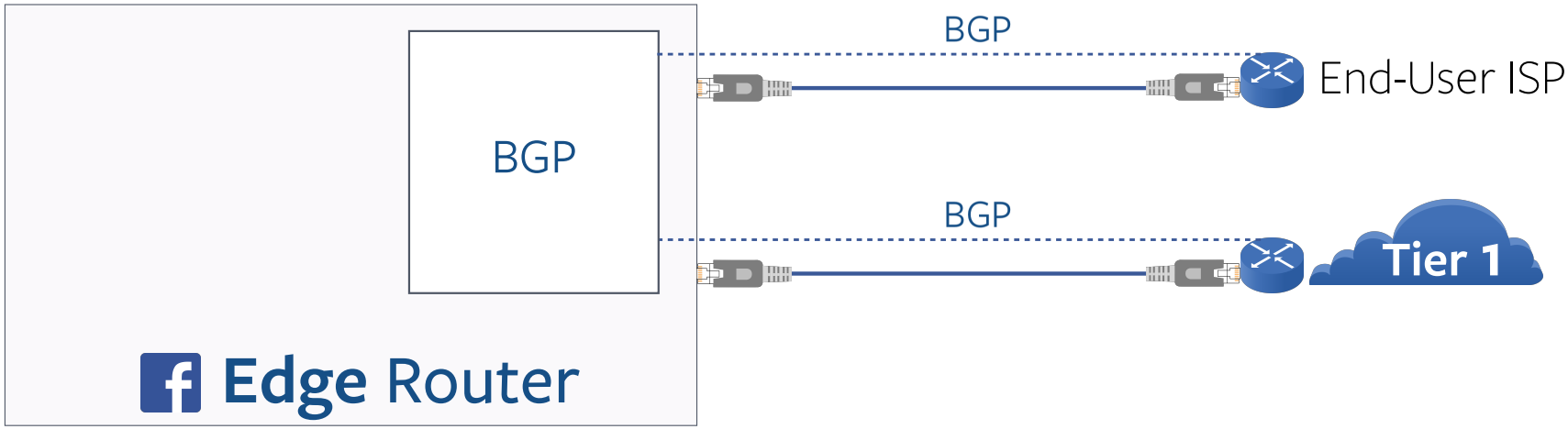

1 | Establish physical circuits

# Basics of Interconnection at a PoP




1 | Establish physical circuits

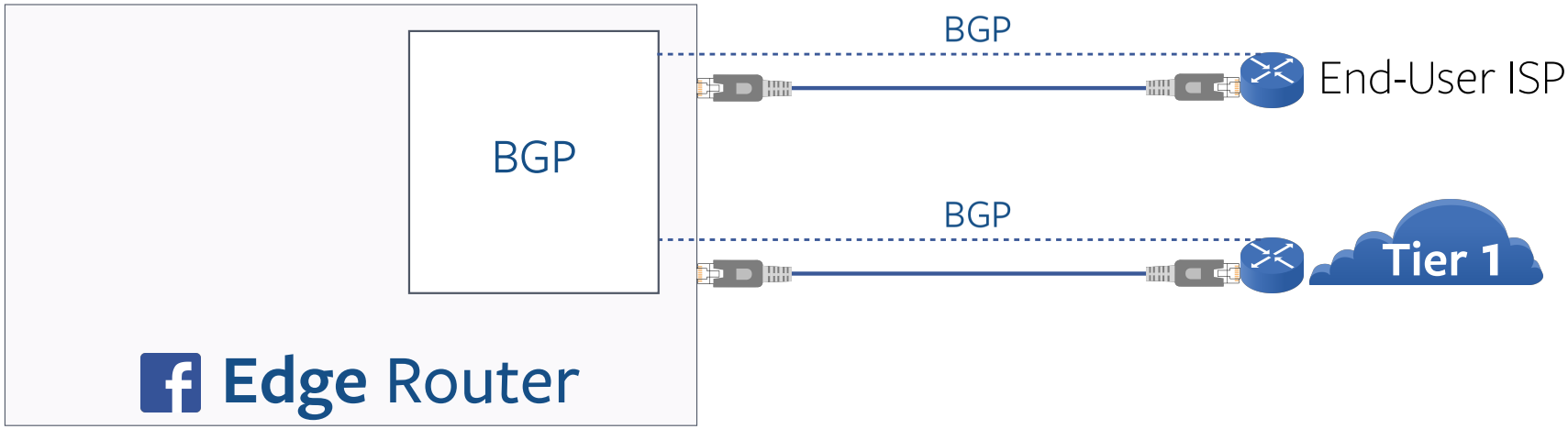
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

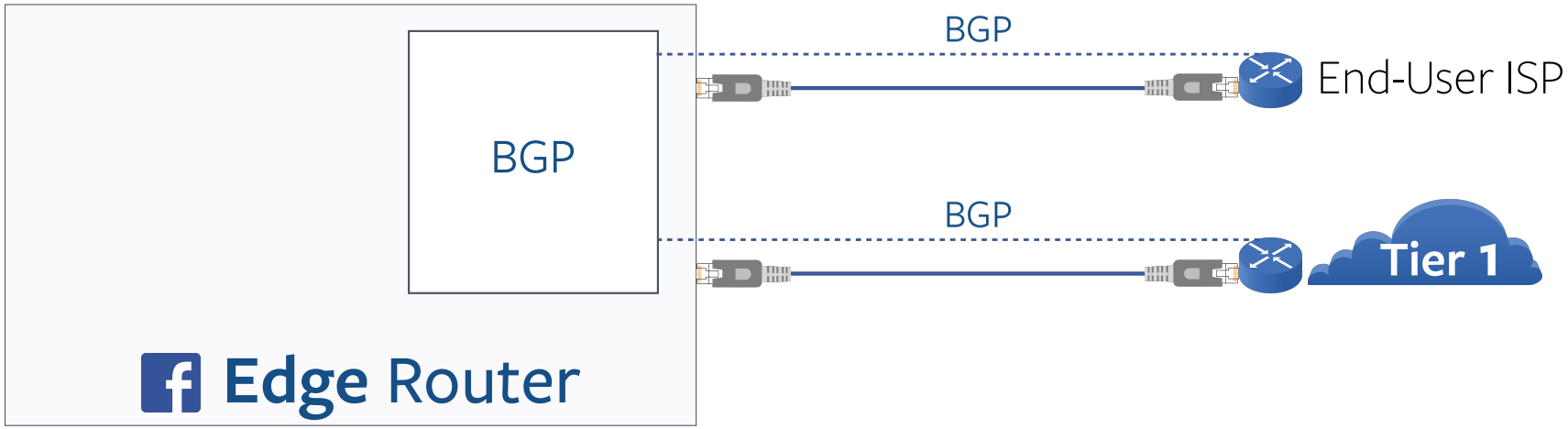
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

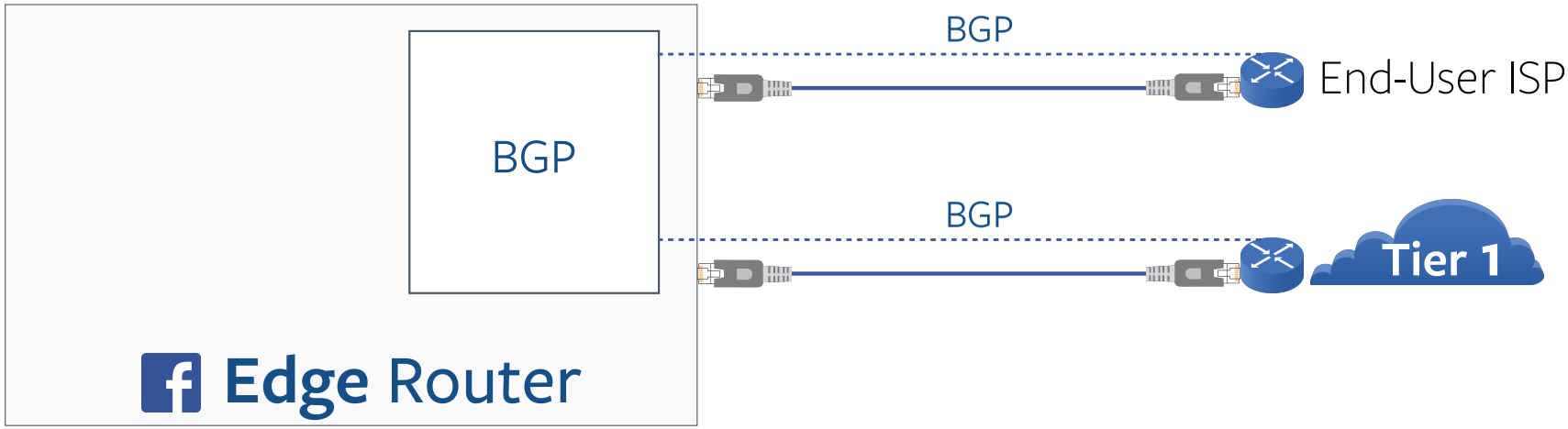
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

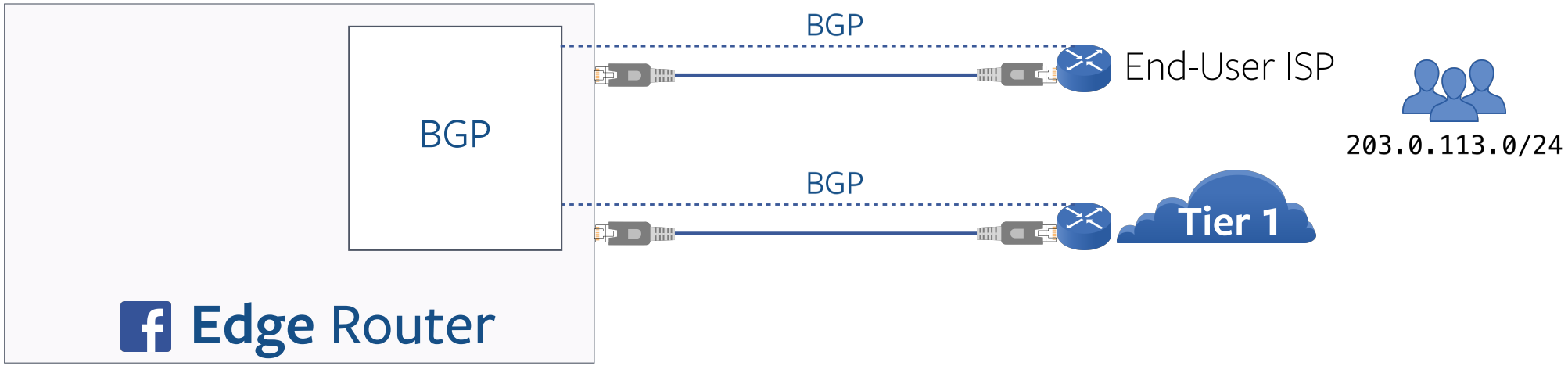
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

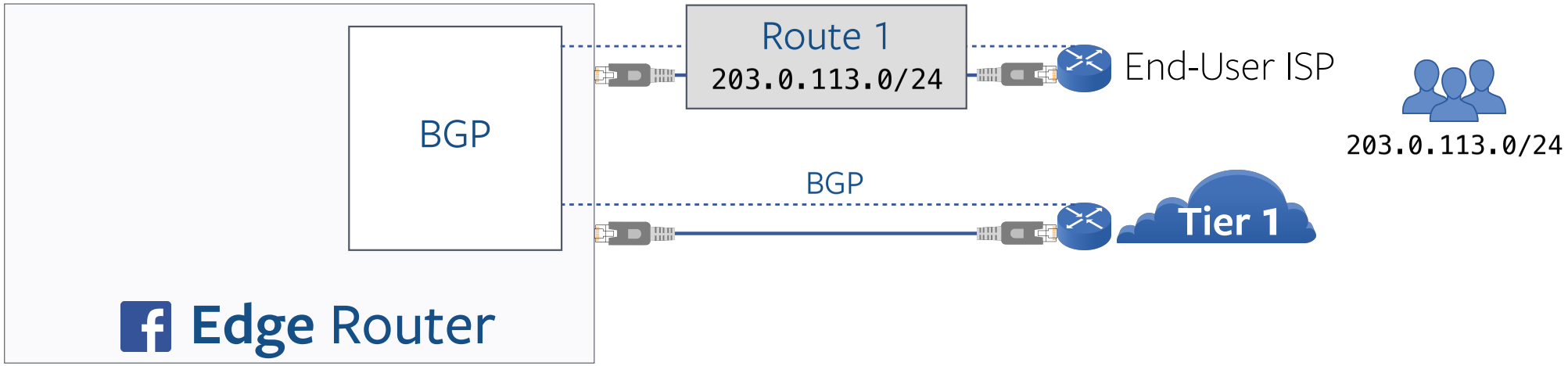
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

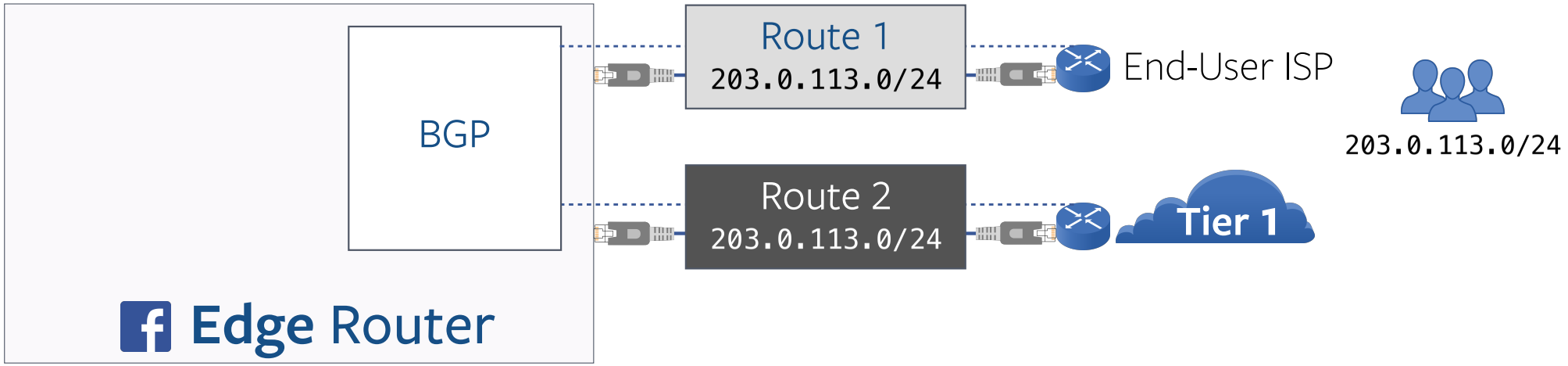
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

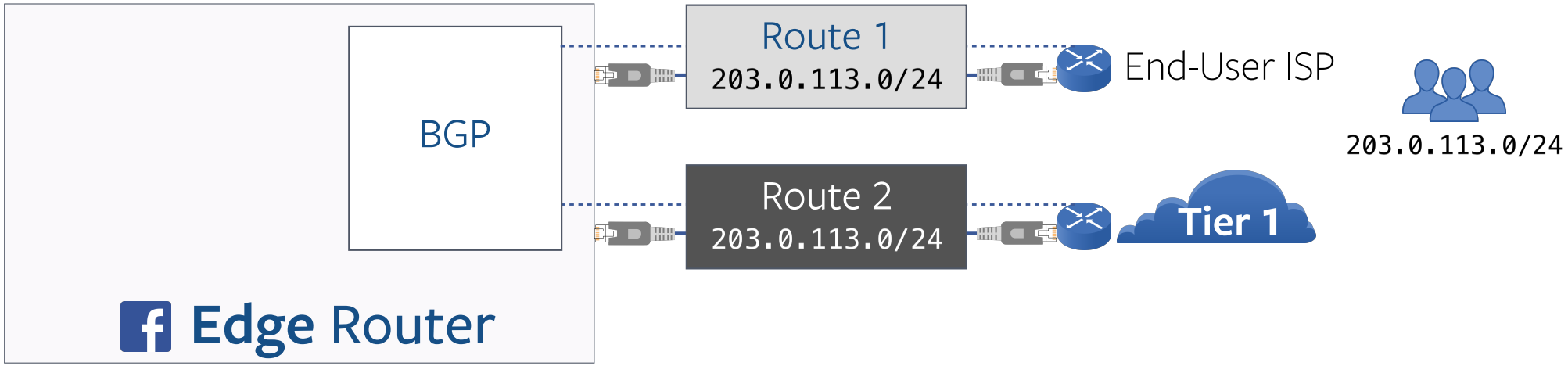
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

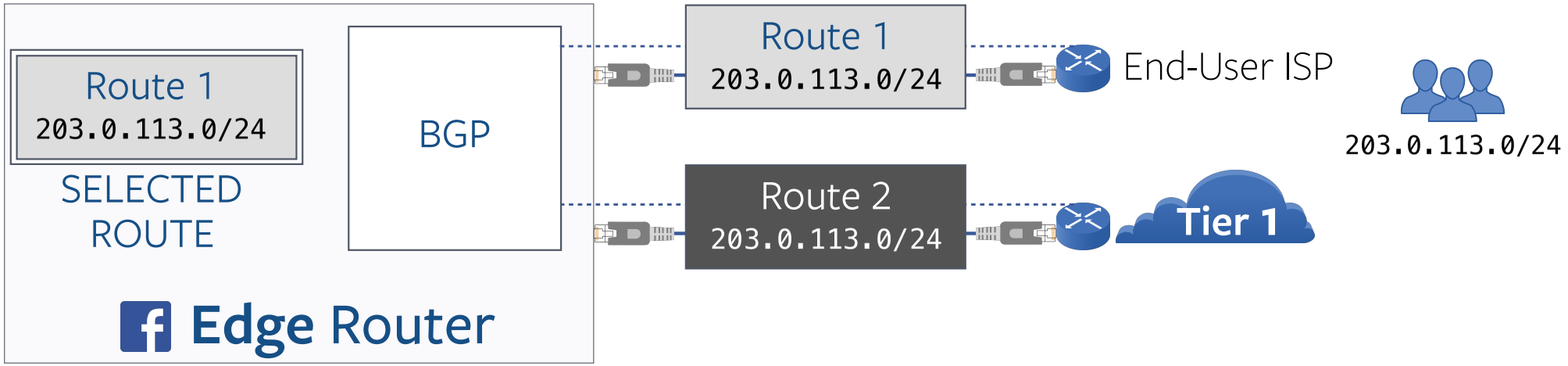
# Basics of Interconnection at a PoP




1 Establish physical circuits

2 Exchange reachability information via BGP BGP

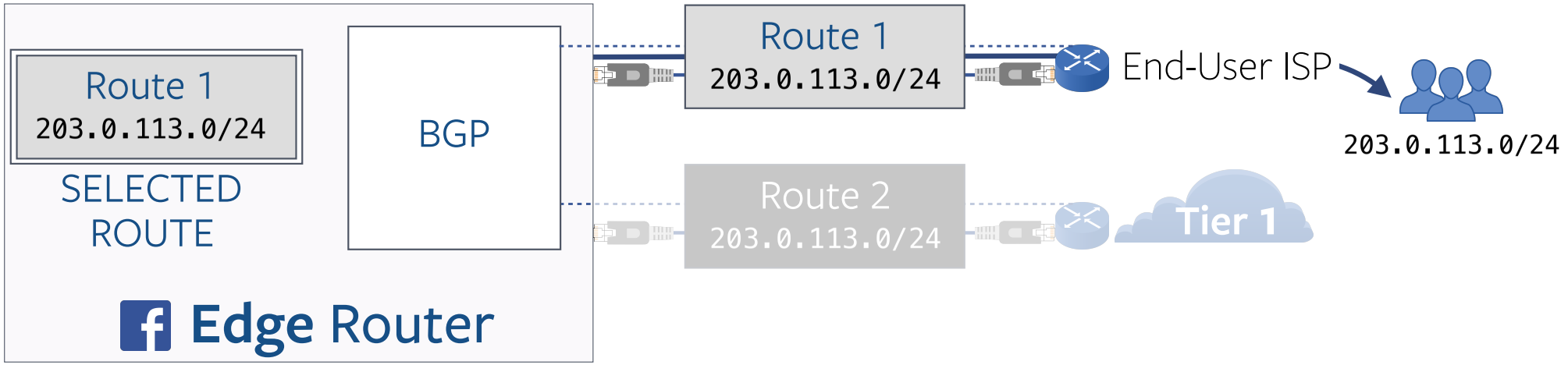
# Basics of Interconnection at a PoP




1. Establish physical circuits
2. Exchange reachability information via BGP
3. BGP at router selects which route to use

BGP

# Basics of Interconnection at a PoP




1. Establish physical circuits
2. Exchange reachability information via BGP
3. BGP at router selects which route to use

BGP

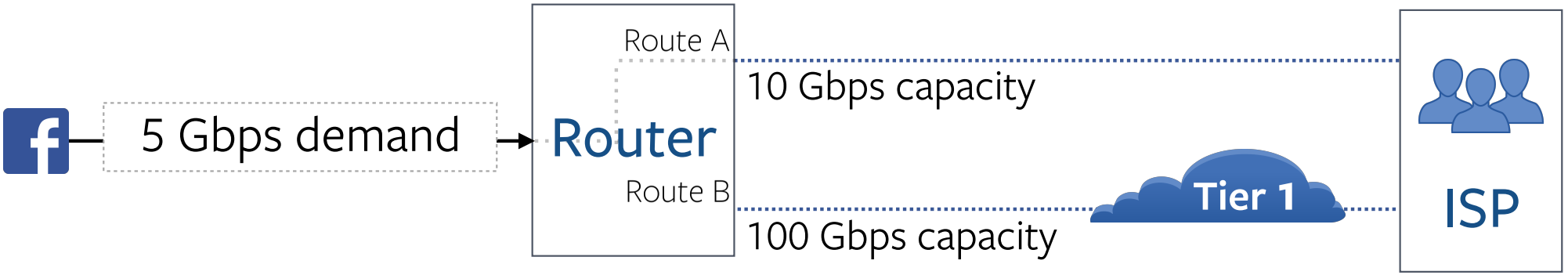
# Challenges to Using Our Connectivity

| | |
|---|---|
| objective | deliver traffic with the best performance possible |

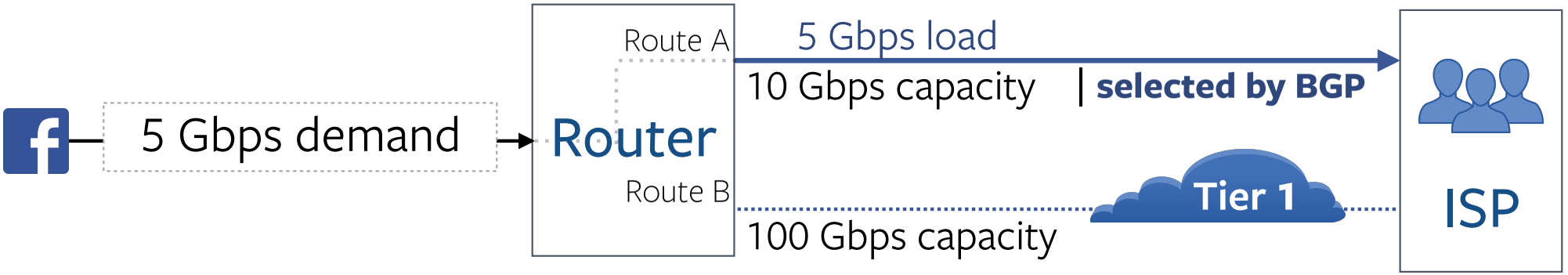
# Challenges to Using Our Connectivity

| | |
|---|---|
| objective | deliver traffic with the best performance possible |
| challenge | BGP does not consider demand, capacity or performance |

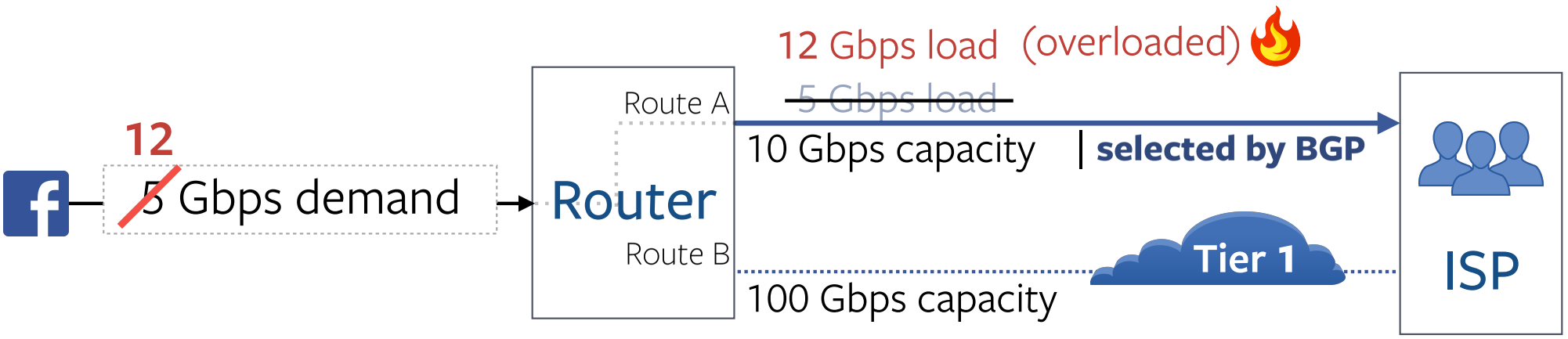
# BGP Does Not Consider Demand and Capacity



# BGP Does Not Consider Demand and Capacity



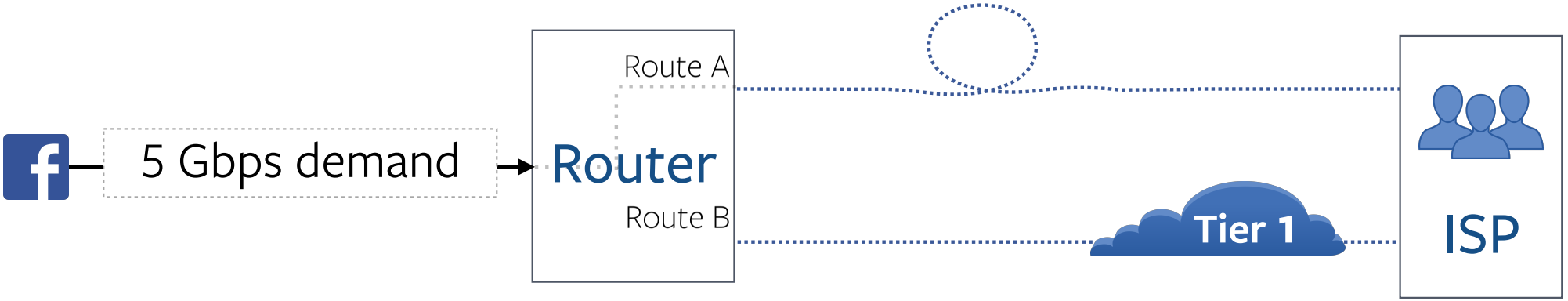
# BGP Does Not Consider Demand and Capacity



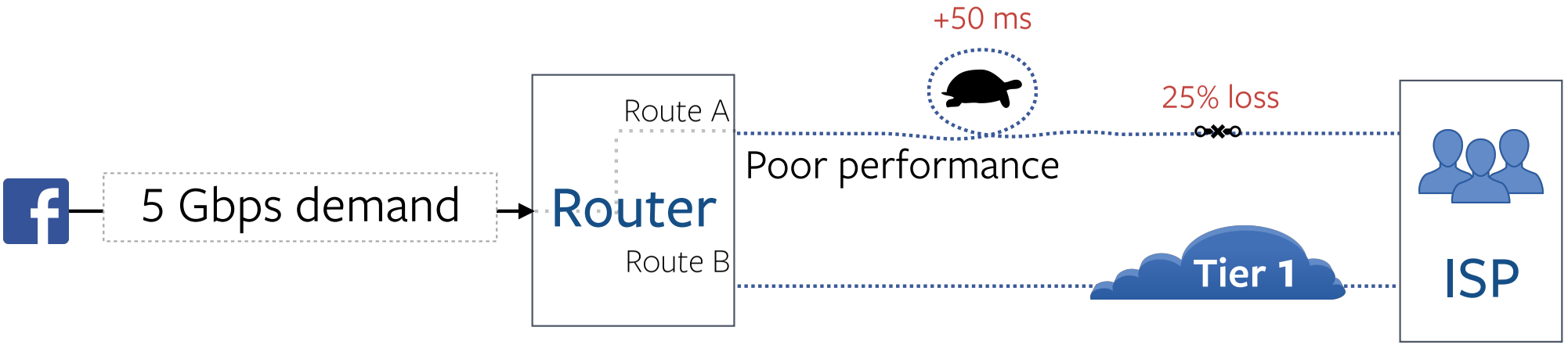

Cannot configure BGP to adapt to demand/capacity in real time

Not possible to express with BGP policy terms

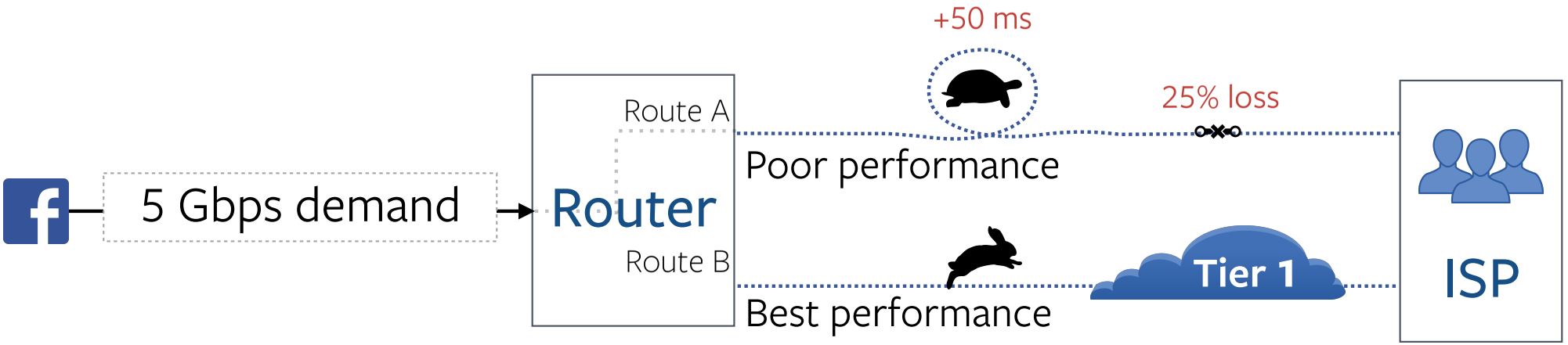
# BGP Does Not Consider Performance



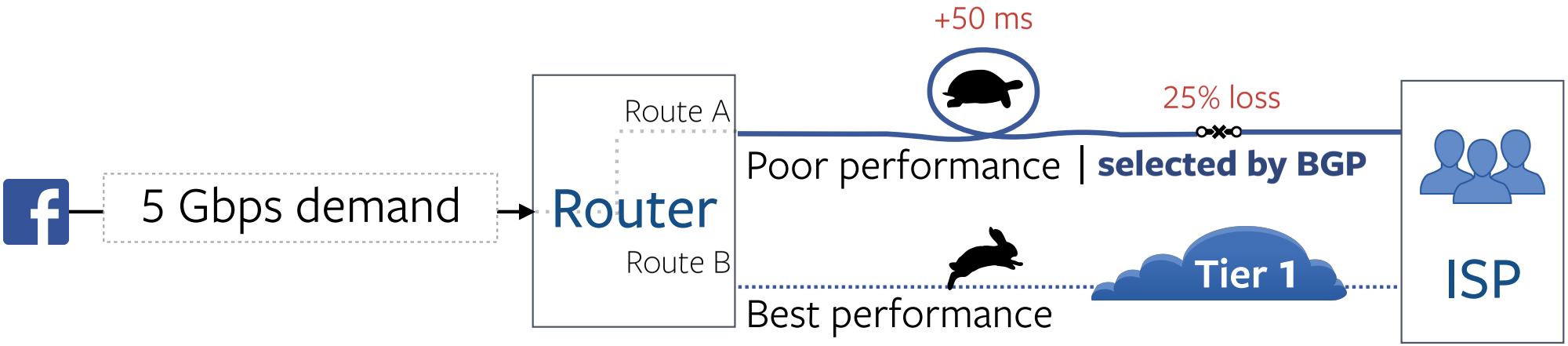
# BGP Does Not Consider Performance



# BGP Does Not Consider Performance



# BGP Does Not Consider Performance




Cannot configure BGP to adapt to performance in real time

Not possible to express with BGP policy terms

BGP is fundamental to interconnection

and it's not going away

# Sidestepping BGP's Limitations

| | |
|---|---|
| objective | deliver traffic with the best performance possible |
| challenge | BGP does not consider demand, capacity or performance |
| approach | shift control from BGP at routers to a software controller |

# Outline

# Outline

1 Overview

2 Facebook's Connectivity and Challenges

# Outline

1 Overview

2 Facebook's Connectivity and Challenges

3 Sidestepping BGP's Limitations with Edge Fabric

# Outline

1. Overview
2. Facebook's Connectivity and Challenges
3. Sidestepping BGP's Limitations with Edge Fabric
4. Results from Edge Fabric's Behavior in Production

# Outline

# Connectivity at a Point of Presence (PoP)

## Transit Providers

deliver traffic to entire Internet

| # per PoP | Interconnection |
| --- | --- |
| Two or more | Private circuit |

# Connectivity at a Point of Presence (PoP)

| | # per PoP | Interconnection |
|---|---|---|
| Transit Providers<br>deliver traffic to entire Internet | Two or more | Private circuit |
| Peers<br>end-user ISPs, mobile providers | | |
| Private Peers | Tens | Private circuit |

# Connectivity at a Point of Presence (PoP)

| | # per PoP | Interconnection |
|---|---|---|
| Transit Providers<br>deliver traffic to entire Internet | Two or more | Private circuit |
| Peers<br>end-user ISPs, mobile providers | | |
| Private Peers | Tens | Private circuit |
| IXP Peers<br>via Internet Exchange Point | Hundreds | Shared fabric |

# Connectivity at a Point of Presence (PoP)

# **Connectivity at a Point of Presence** (PoP)

We prefer routes from
private peers > IXP peers > transits

# **Connectivity at a Point of Presence** (PoP)

We prefer routes from
**private peers > IXP peers > transits**

peers > transits | peers provide short, direct paths to end users

# **Connectivity at a Point of Presence** (PoP)

We prefer routes from
**private peers > IXP peers > transits**

| | |
|---|---|
| peers > transits | peers provide short, direct paths to end users |
| private > IXP peers | prefer circuits dedicated to Facebook and peer |

# Connectivity at a Point of Presence (PoP)



Transit Providers

deliver traffic to entire Internet

Peers

end-user ISPs, mobile providers

Private Peers

majority of traffic



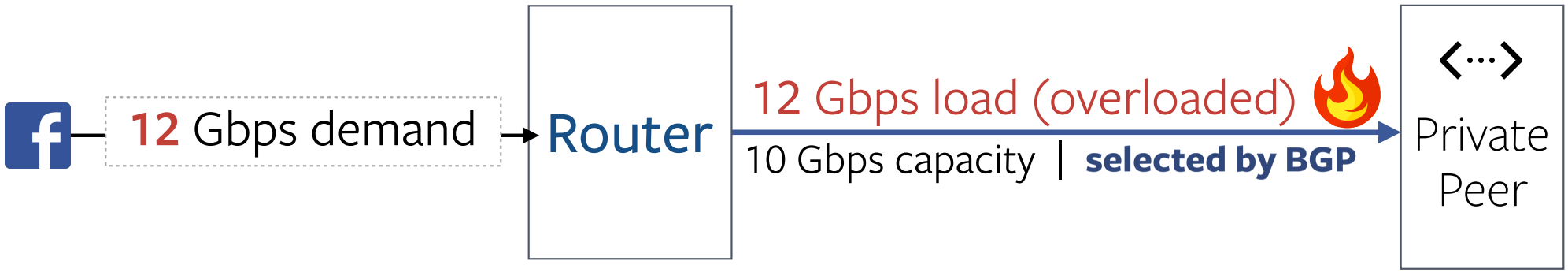
IXP Peers

via Internet Exchange Point

| | # per PoP | Interconnection |
|---|---|---|
| Transit Providers | Two or more | Private circuit |
| Peers | | |
| Private Peers | Tens | Private circuit |
| IXP Peers | Hundreds | Shared fabric |

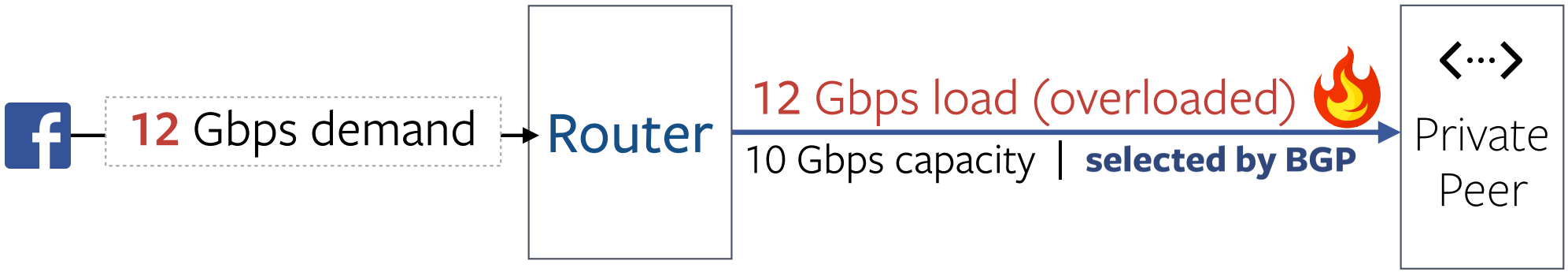
# Capacity Constraints in Production

We cannot acquire **sufficient capacity** with private peers to satisfy demand



# Capacity Constraints in Production

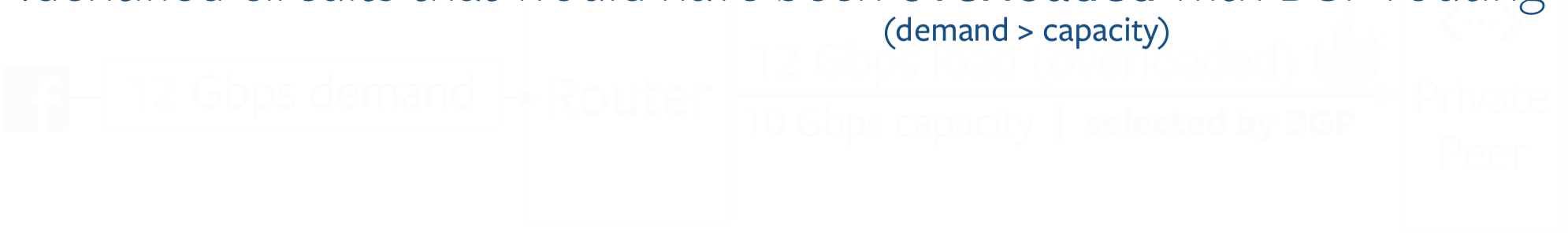
We cannot acquire **sufficient capacity** with private peers to satisfy demand



# Capacity Constraints in Production

We cannot acquire **sufficient capacity** with private peers to satisfy demand

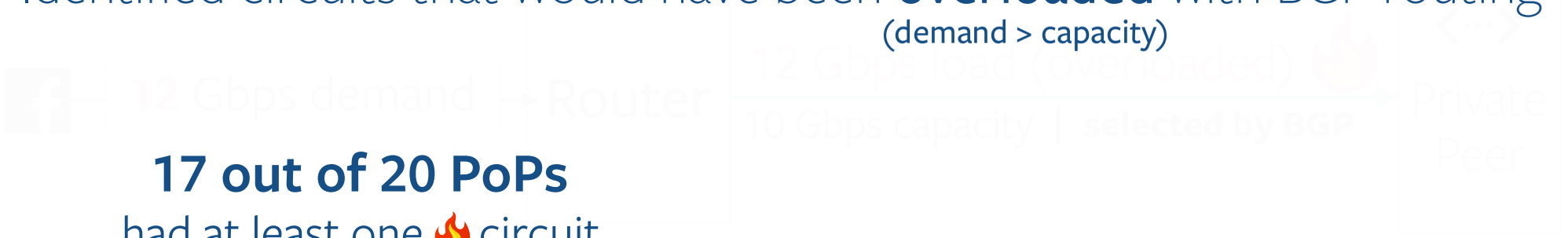
**Two-day study of 20 PoPs:**
Identified circuits that would have been **overloaded** with BGP routing
(demand > capacity)



# Capacity Constraints in Production

We cannot acquire **sufficient capacity** with private peers to satisfy demand

**Two-day study of 20 PoPs:**
Identified circuits that would have been **overloaded** with BGP routing
(demand > capacity)

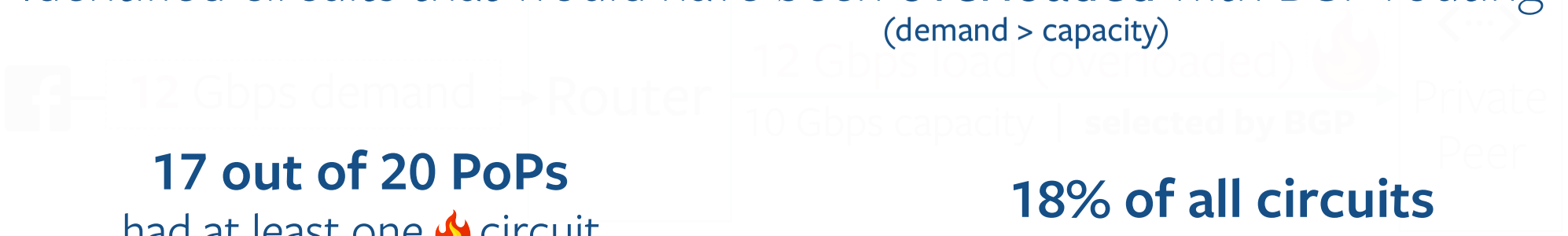



**17 out of 20 PoPs**
had at least one circuit

# Capacity Constraints in Production

We cannot acquire **sufficient capacity** with private peers to satisfy demand

**Two-day study of 20 PoPs:**
Identified circuits that would have been **overloaded** with BGP routing
(demand > capacity)

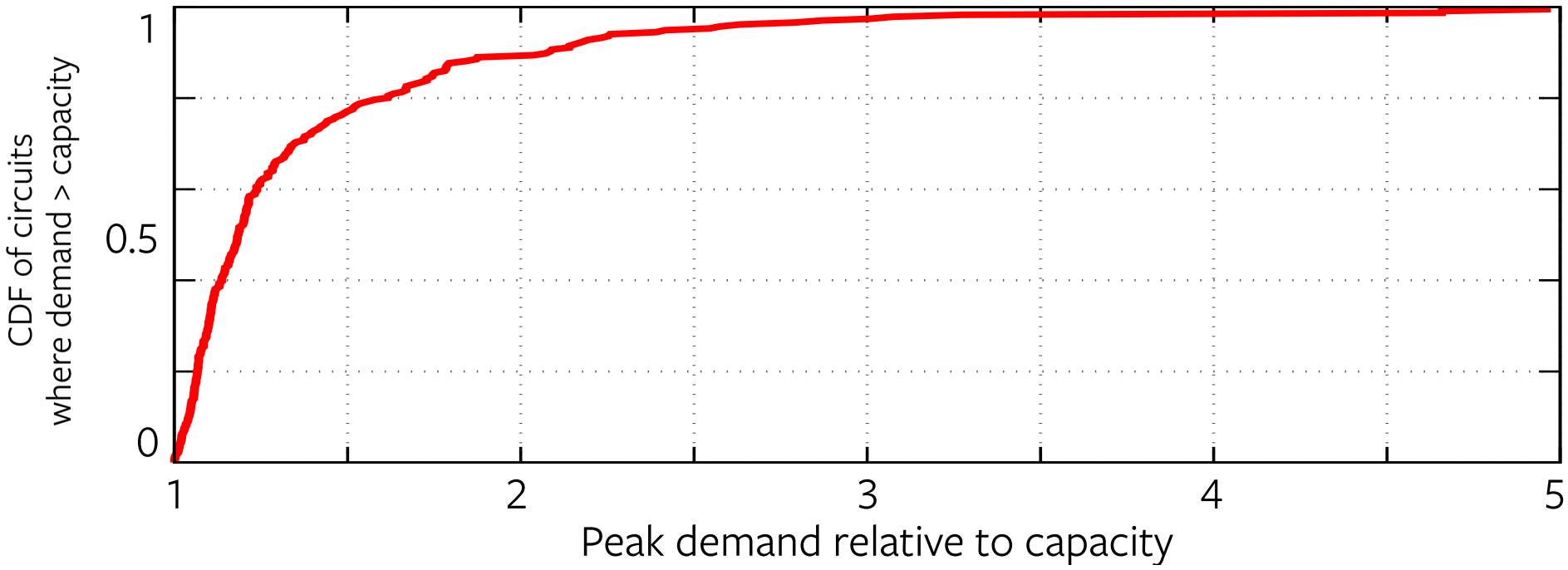



**17 out of 20 PoPs**
had at least one circuit

**18% of all circuits**

# Capacity Constraints in Production

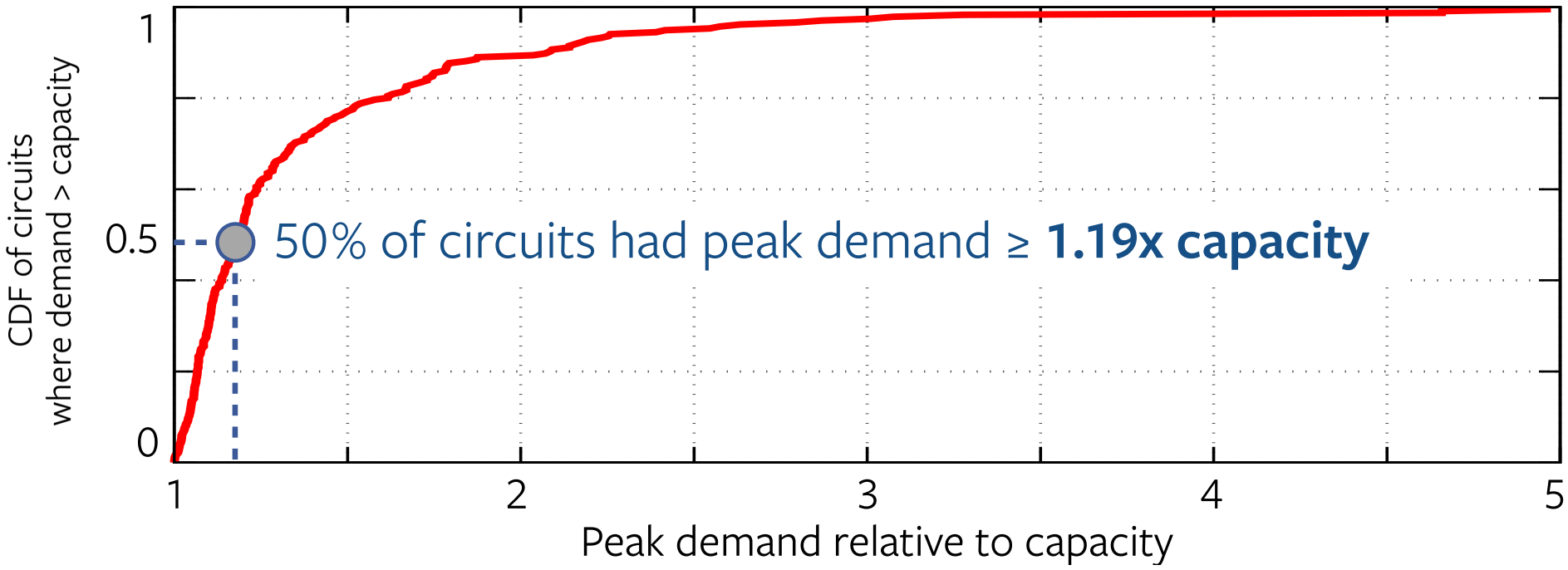
# Capacity Constraints in Production




## Circuit's peak demand to capacity

For circuits predicted to have demand > capacity at least once

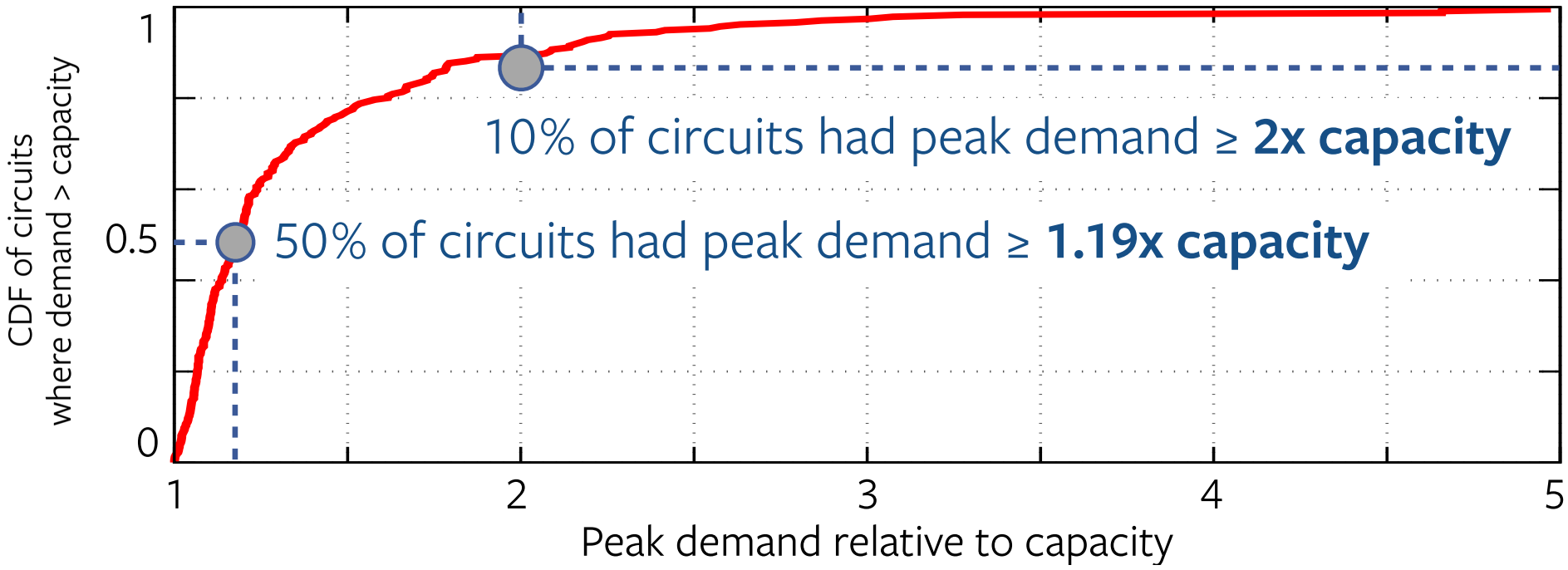
# Capacity Constraints in Production




## Circuit's peak demand to capacity

For circuits predicted to have demand > capacity at least once

# Capacity Constraints in Production



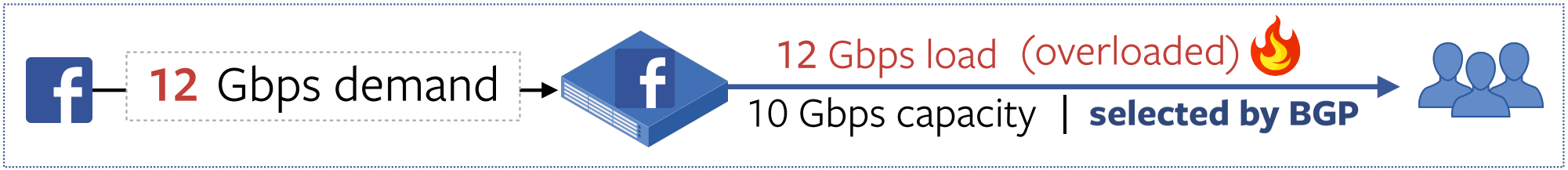

## Circuit's peak demand to capacity

For circuits predicted to have demand > capacity at least once

# Capacity Constraints in Production

BGP does not consider demand or capacity

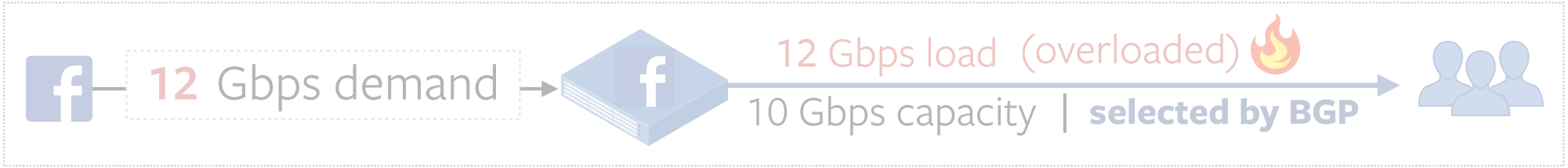
↳ situations where demand > capacity, degrading user experience



# Capacity Constraints in Production

BGP does not consider demand or capacity

↳ situations where demand > capacity, degrading user experience




BGP's decision process doesn't meet our needs

so we built **Edge Fabric**

# Outline

# Sidestepping BGP's Limitations

| | |
|---|---|
| objective | deliver traffic with the best performance possible |
| challenge | BGP does not consider demand, capacity or performance |
| approach | shift control from BGP at routers to a software controller |

# Design Priorities

## Operational simplicity

minimize change and system complexity

# Design Priorities

## Operational simplicity

minimize change and system complexity

## Ease of deployment

interoperate with existing infrastructure and tooling

# Responsibility for Routing

design priorities

Operational simplicity

Ease of deployment

**Traditional routers**

Route per destination from BGP

**Host-based routing**

Route per packet dictated by hosts

# Responsibility for Routing

design priorities

Operational simplicity

Ease of deployment

**Traditional routers**

Route per destination from BGP

**Host-based routing**

Route per packet dictated by hosts

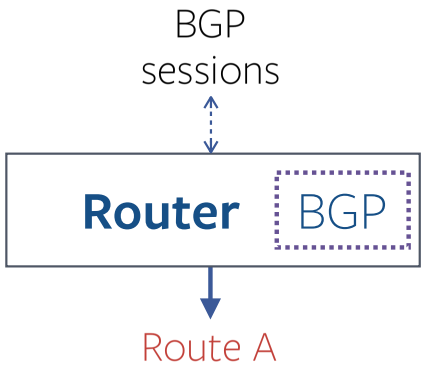
## Edge Fabric's approach:

Controller overrides BGP's decisions at router

Hosts provide hints on packet priority

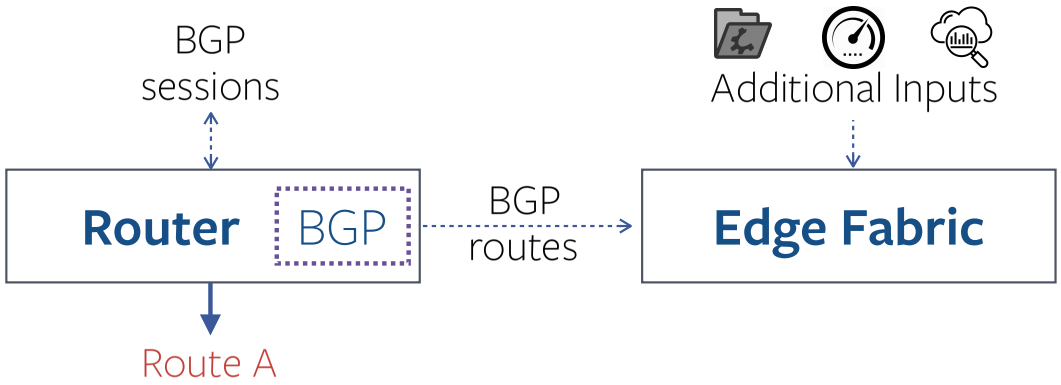
# Edge Fabric's Approach to Control

1 Router selects routes using BGP



# Edge Fabric's Approach to Control

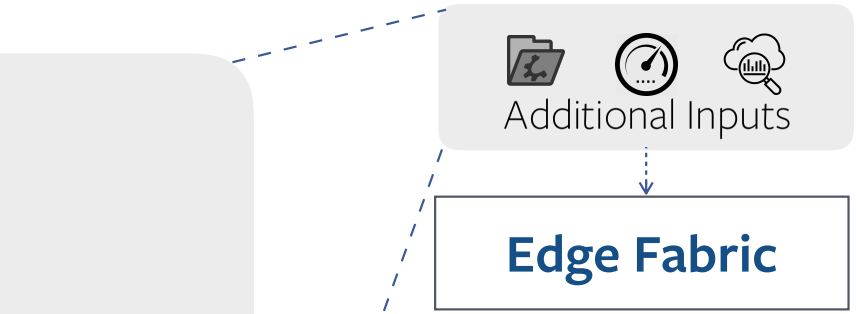
1. Router selects routes using BGP
2. **Edge Fabric** selects ideal routes using BGP routes + other inputs



# Edge Fabric's Approach to Control

## Inputs to Edge Fabric

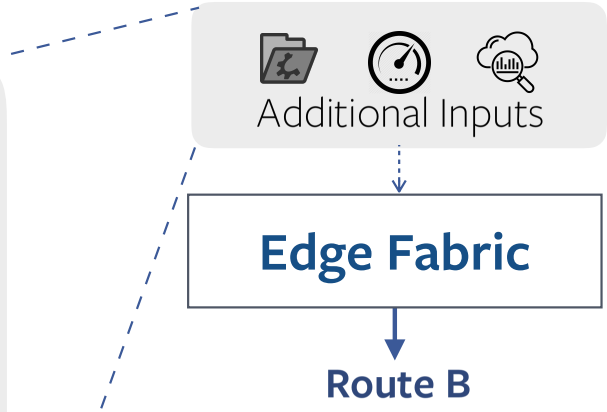
- BGP routes (from router)
- Advanced policy
- **1 Gbps** Prefix traffic rates
- **40 Gbps** Circuit capacities
- Route performance measurements



# Edge Fabric's Approach to Control

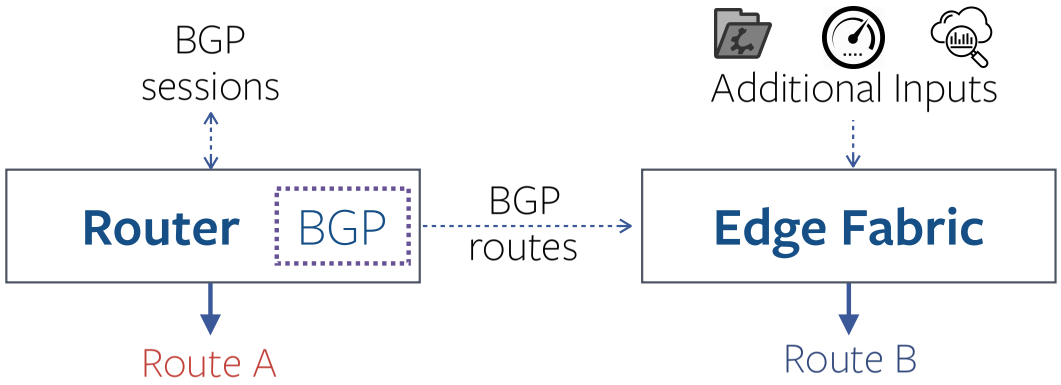
## Inputs to Edge Fabric

- BGP routes (from router)
- Advanced policy
- 1 Gbps Prefix traffic rates
- 40 Gbps Circuit capacities
- Route performance measurements



# Edge Fabric's Approach to Control

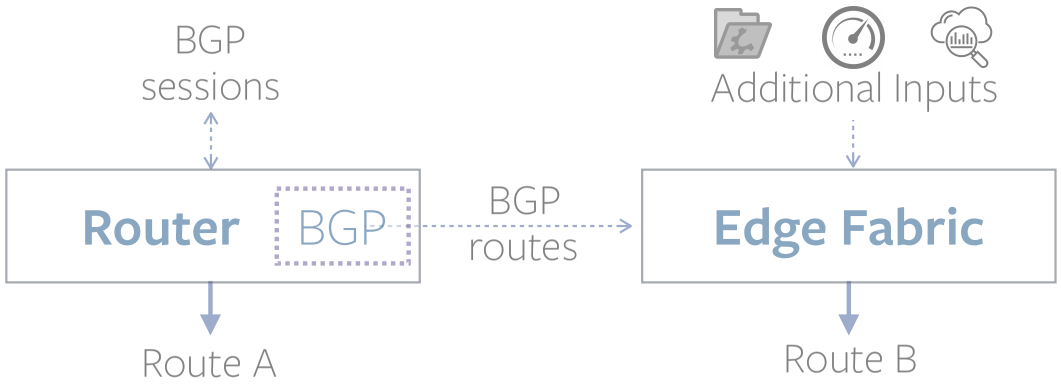
1. Router selects routes using BGP
2. **Edge Fabric** selects ideal routes using BGP routes + other inputs



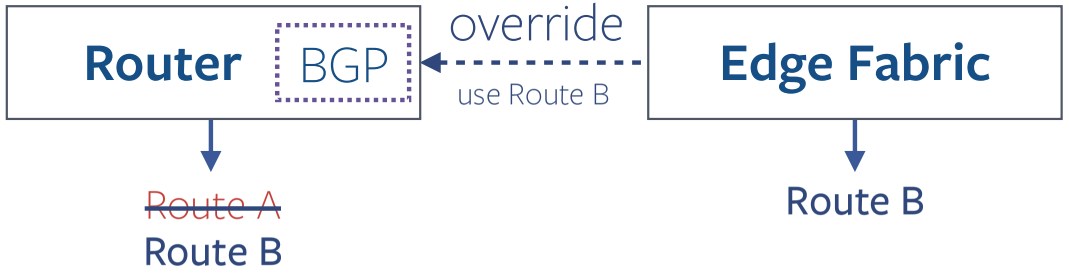
# Edge Fabric's Approach to Control

1 Router selects routes using BGP

2 **Edge Fabric** selects ideal routes using BGP routes + other inputs




3 If router and Edge Fabric choose different routes, **override router**



# Types of Edge Fabric Overrides

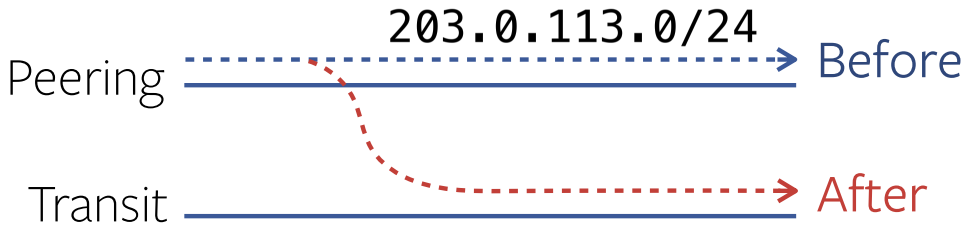
## Edge Fabric can override BGP's decision in order to...

# Types of Edge Fabric Overrides

## Edge Fabric can override BGP's decision in order to...

Move traffic for set of end-users

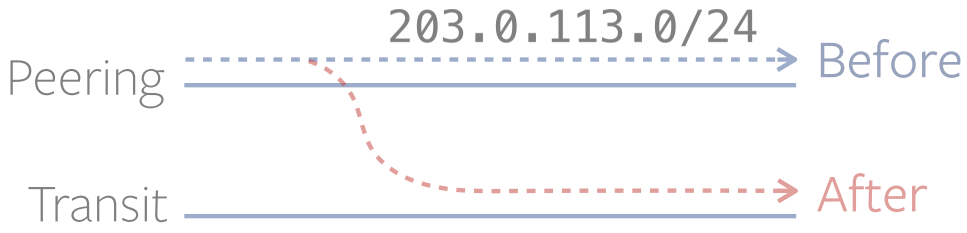
override per <destination>



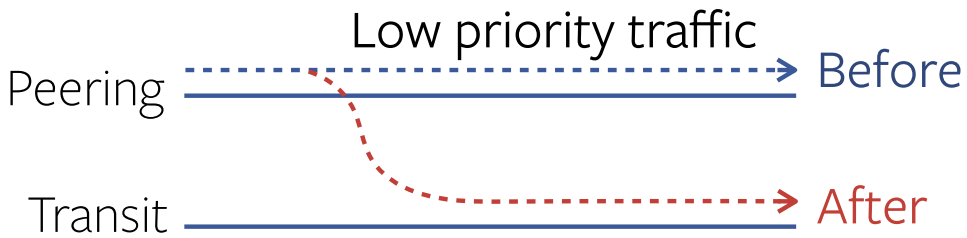
# Types of Edge Fabric Overrides

## Edge Fabric can override BGP's decision in order to...

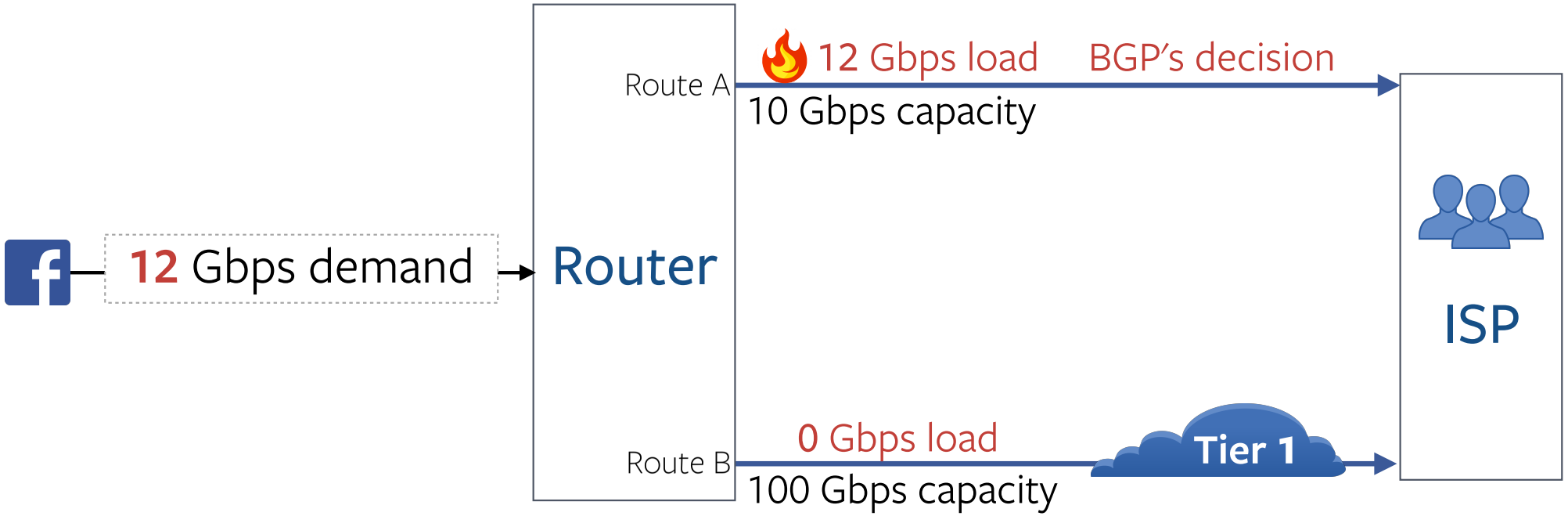
Move traffic for set of end-users
override per <destination>



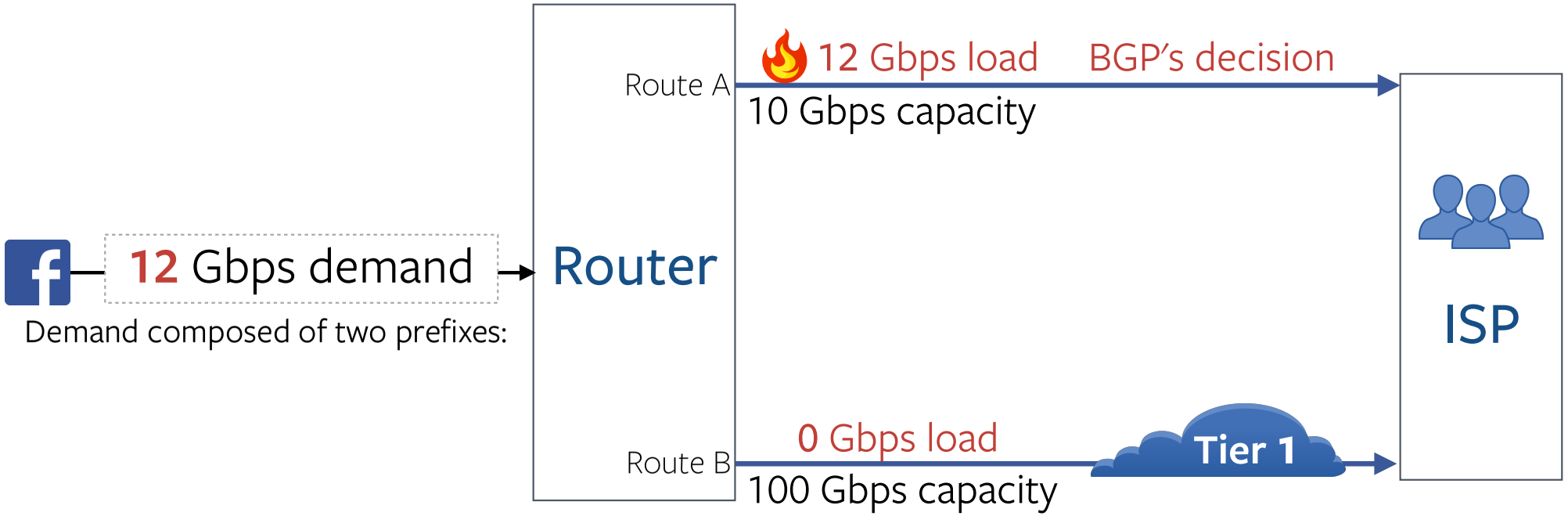

Move class of end-user traffic
override per <destination, traffic class>



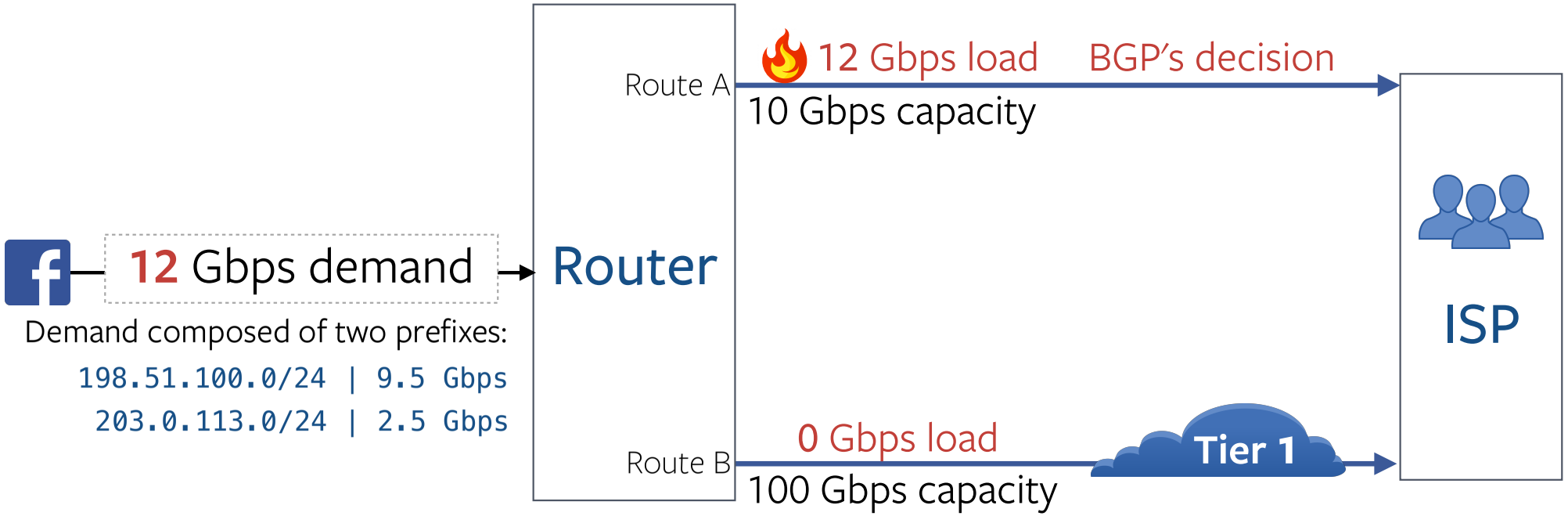
# **Example EF Override:** Preventing Congestion



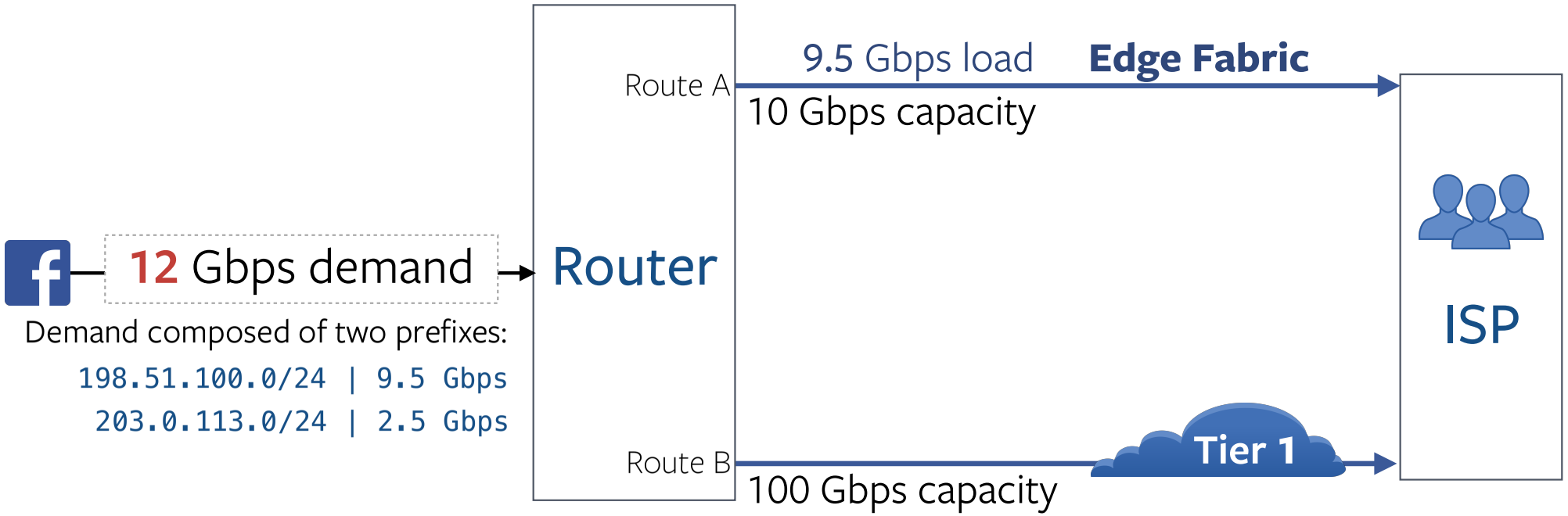
# **Example EF Override:** Preventing Congestion



# **Example EF Override:** Preventing Congestion

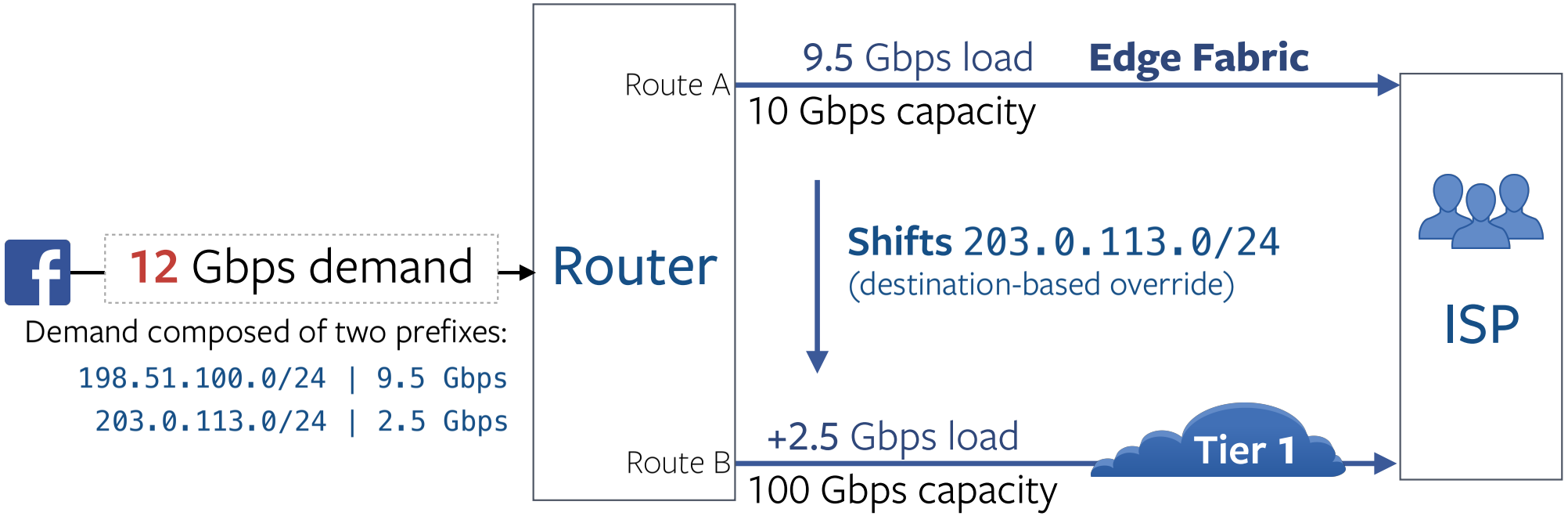


# **Example EF Override:** Preventing Congestion




**Edge Fabric shifts a prefix's traffic to an alternate link**

# **Example EF Override:** Preventing Congestion

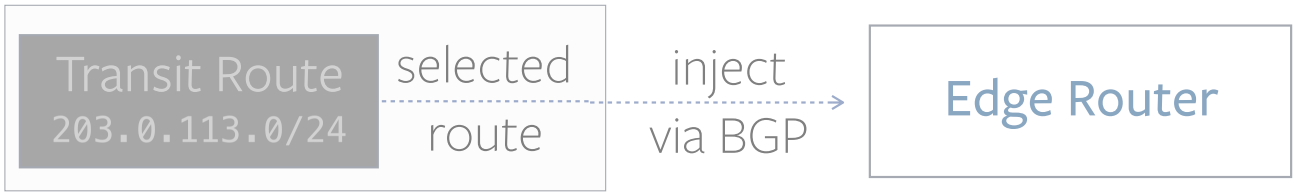



**Edge Fabric shifts a prefix's traffic to an alternate link**

# **Example EF Override:** Preventing Congestion


1 **Edge Fabric** injects override route via BGP

# **Example EF Override:** Preventing Congestion




1 **Edge Fabric** injects override route via BGP


2 BGP at routers prefers routes from **Edge Fabric**

LocalPref used to prefer routes injected by Edge Fabric

# **Example EF Override:** Preventing Congestion

**Edge Fabric** monitors BGP's decisions
and overrides as needed to **prevent congestion**

# Edge Fabric Supports Variety of TE Policies

- BGP routes
- Policy
- Circuit capacity and traffic rates
- Route performance measurements

inputs

- Path per <destination>
- Path per <destination, traffic class>

override granularities

**Edge Fabric** supports sophisticated traffic engineering policies

# Edge Fabric Supports Variety of TE Policies

- BGP routes
- Policy
- Circuit capacity and traffic rates
- Route performance measurements

inputs

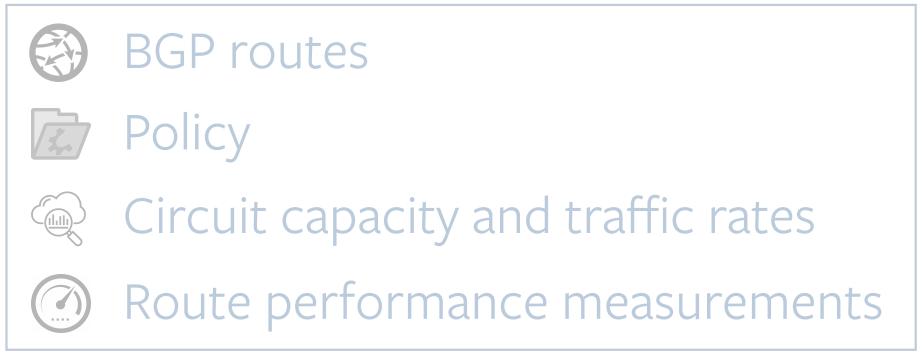
- Path per <destination>
- Path per <destination, traffic class>

override granularities

**Edge Fabric** supports sophisticated traffic engineering policies

**and** is compatible with existing BGP infrastructure

# Edge Fabric Supports Variety of TE Policies




inputs


override granularities

**Edge Fabric** supports sophisticated traffic engineering policies

**and** is compatible with existing BGP infrastructure

**Edge Fabric**: centralized control over distributed BGP process

# Edge Fabric Meets Our Design Priorities

## Operational simplicity

Can fallback to BGP at routers

Allows operators to continue to use existing tools

Synchronization is only required between **Edge Fabric** and routers

# Edge Fabric Meets Our Design Priorities

## Operational simplicity

Can fallback to BGP at routers

Allows operators to continue to use existing tools

Synchronization is only required between **Edge Fabric** and routers

## Ease of deployment

BGP sessions with external peers remain at routers

Uses BGP protocol for injections

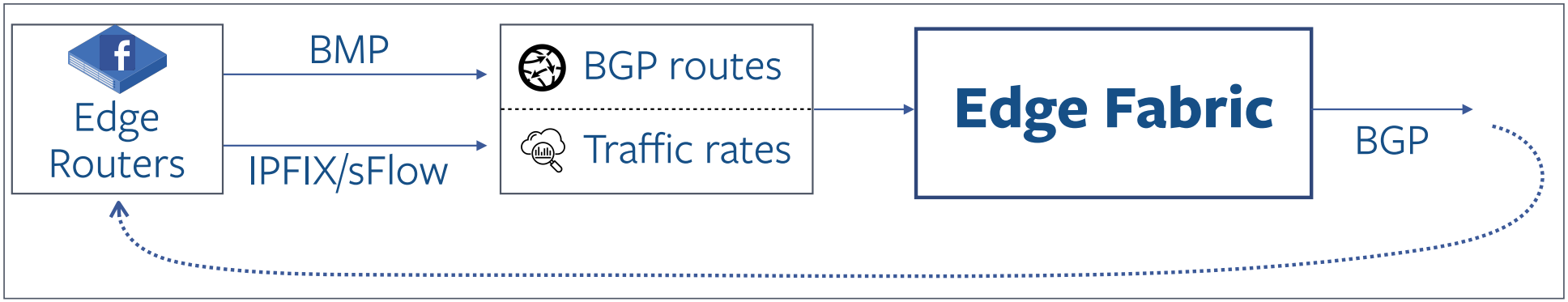
Uses other industry standards for route and traffic info (BMP/IPFIX/sFlow)

# Outline

**Edge Fabric** entered production in 2013

**Objective:** Prevent circuit congestion

# Edge Fabric in Production

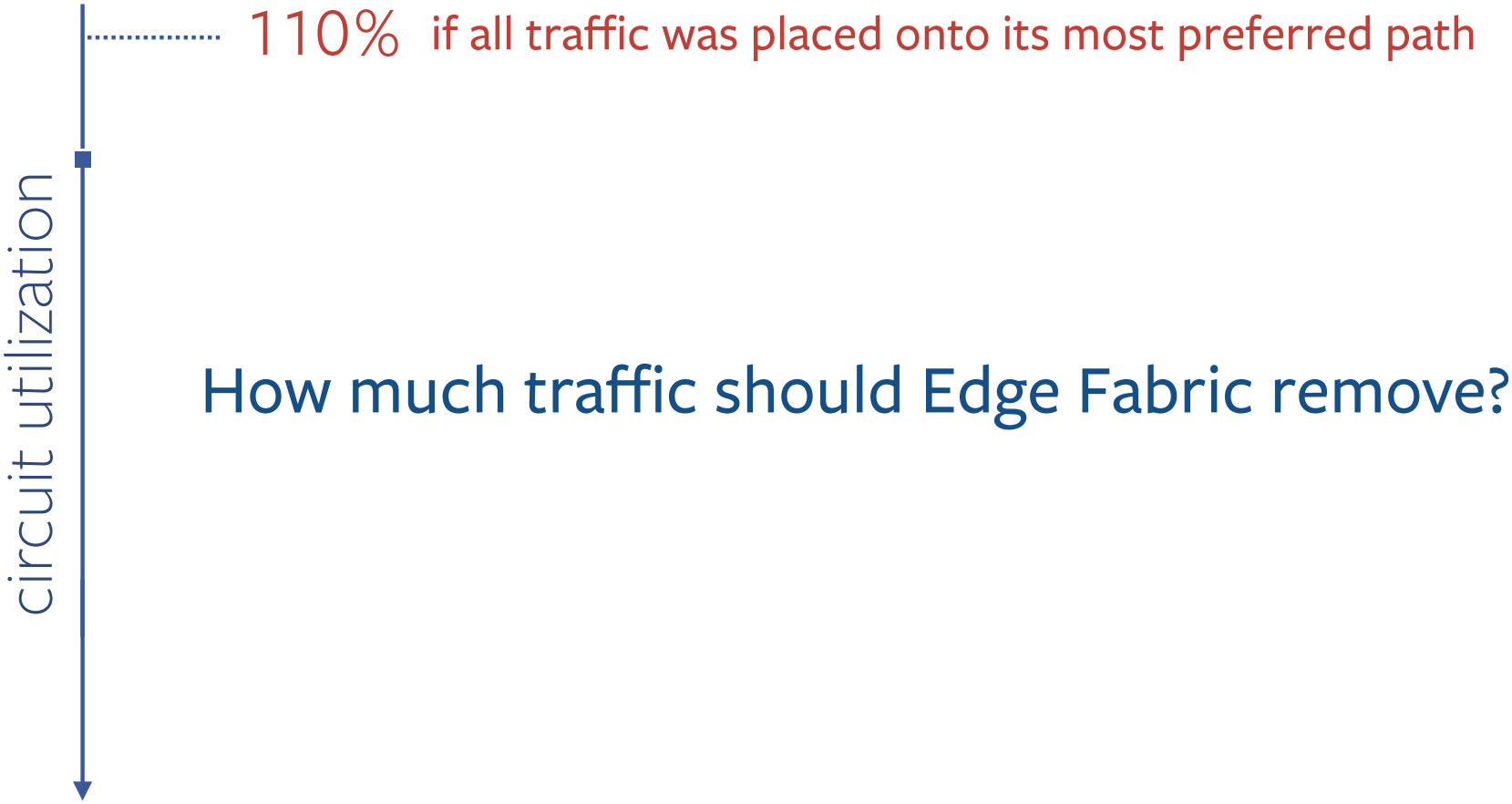



**Runs per PoP, executes every 30 seconds**

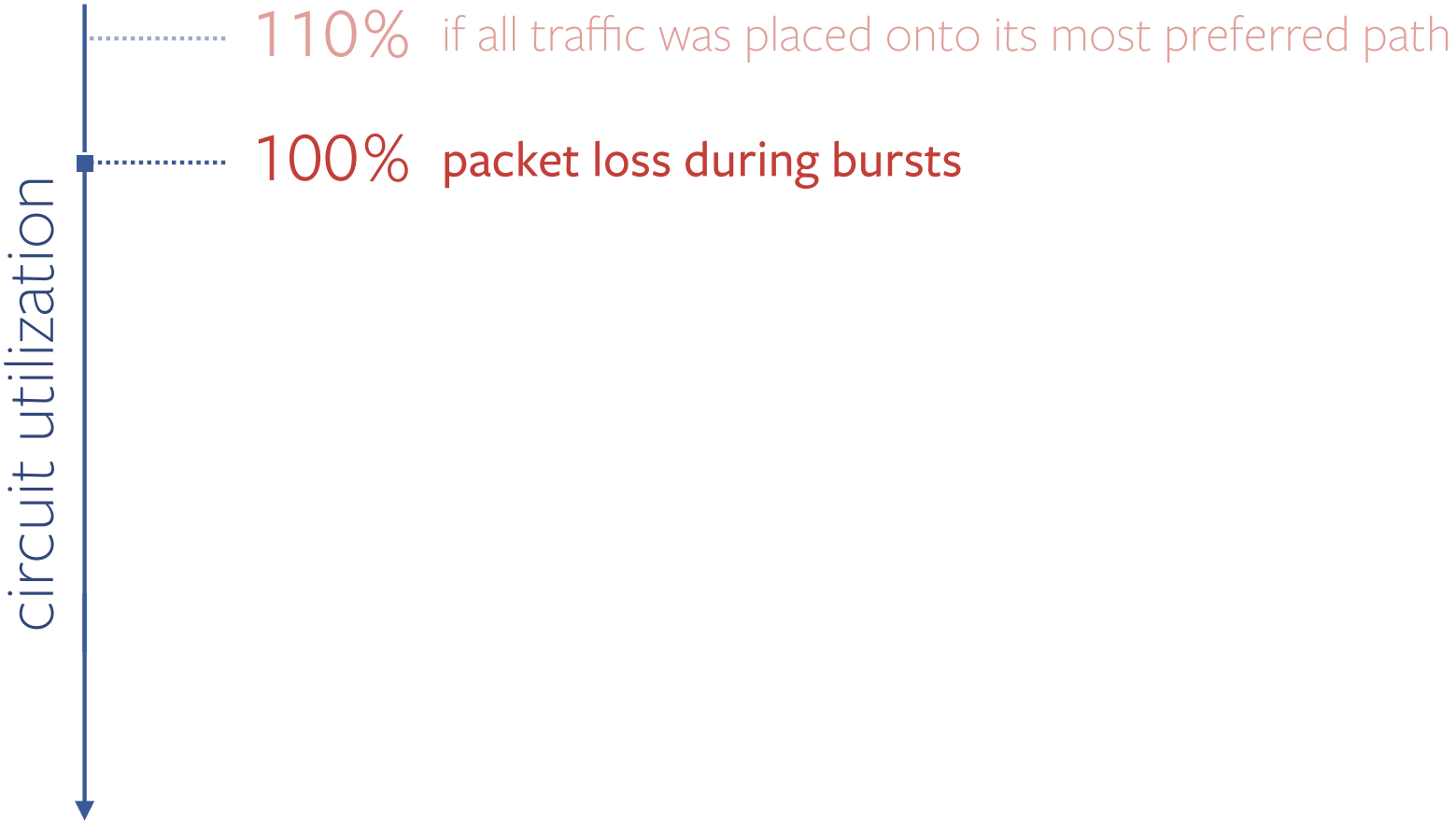
**Controls 100% of Facebook's egress traffic**

(see paper for implementation details)

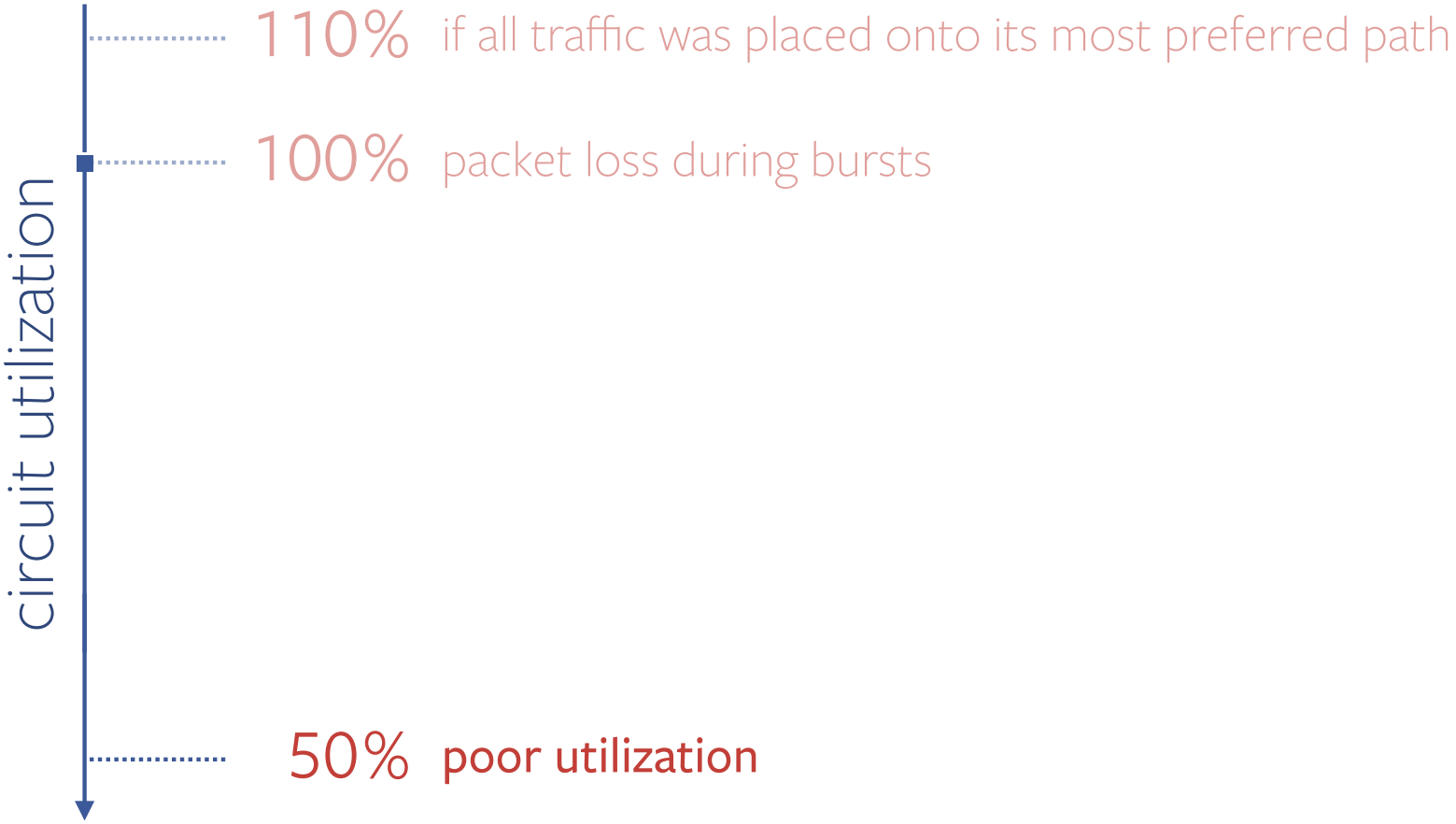
# Target Circuit Utilization To Avoid Congestion



# Target Circuit Utilization To Avoid Congestion



# Target Circuit Utilization To Avoid Congestion



# Target Circuit Utilization To Avoid Congestion

# Evaluating Congestion Avoidance

## Key questions:

Does Edge Fabric prevent circuit congestion and packet drops?

Does Edge Fabric keep circuit utilization at prescribed threshold?

# Evaluating Congestion Avoidance

**Does Edge Fabric prevent circuit congestion and packet drops?**

During measurement period

When Edge Fabric was shifting traffic away
99.9% of the time, no packet drops

# Evaluating Congestion Avoidance

**Does Edge Fabric prevent circuit congestion and packet drops?**

During measurement period

**When Edge Fabric was shifting traffic away**
99.9% of the time, no packet drops

**When Edge Fabric was not active**
No packet drops

# Evaluating Congestion Avoidance

**Does Edge Fabric prevent circuit congestion and packet drops?**

During measurement period

When Edge Fabric was shifting traffic away
99.9% of the time, no packet drops

When Edge Fabric was not active
No packet drops

**Edge Fabric intervened when needed and prevented circuit congestion**

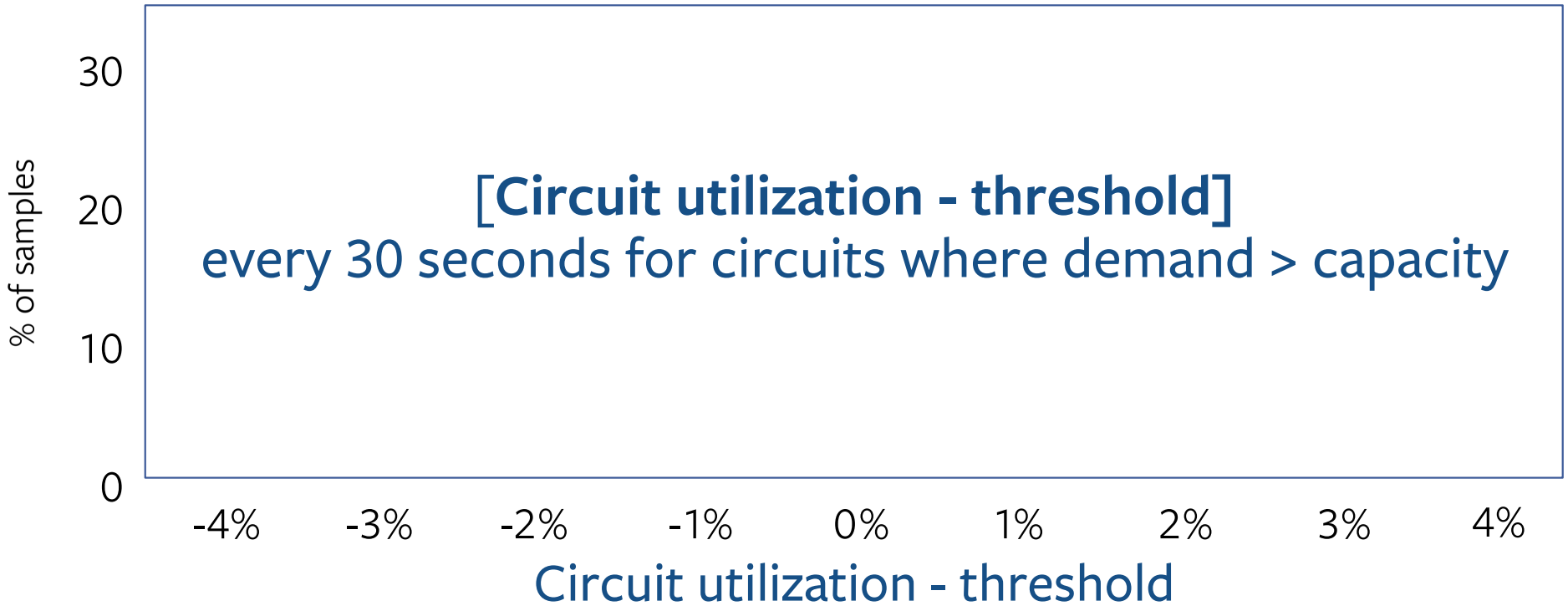
# Evaluating Congestion Avoidance

## Can we keep utilization at the threshold?

[**Circuit utilization - threshold**]
every 30 seconds for circuits where demand > capacity

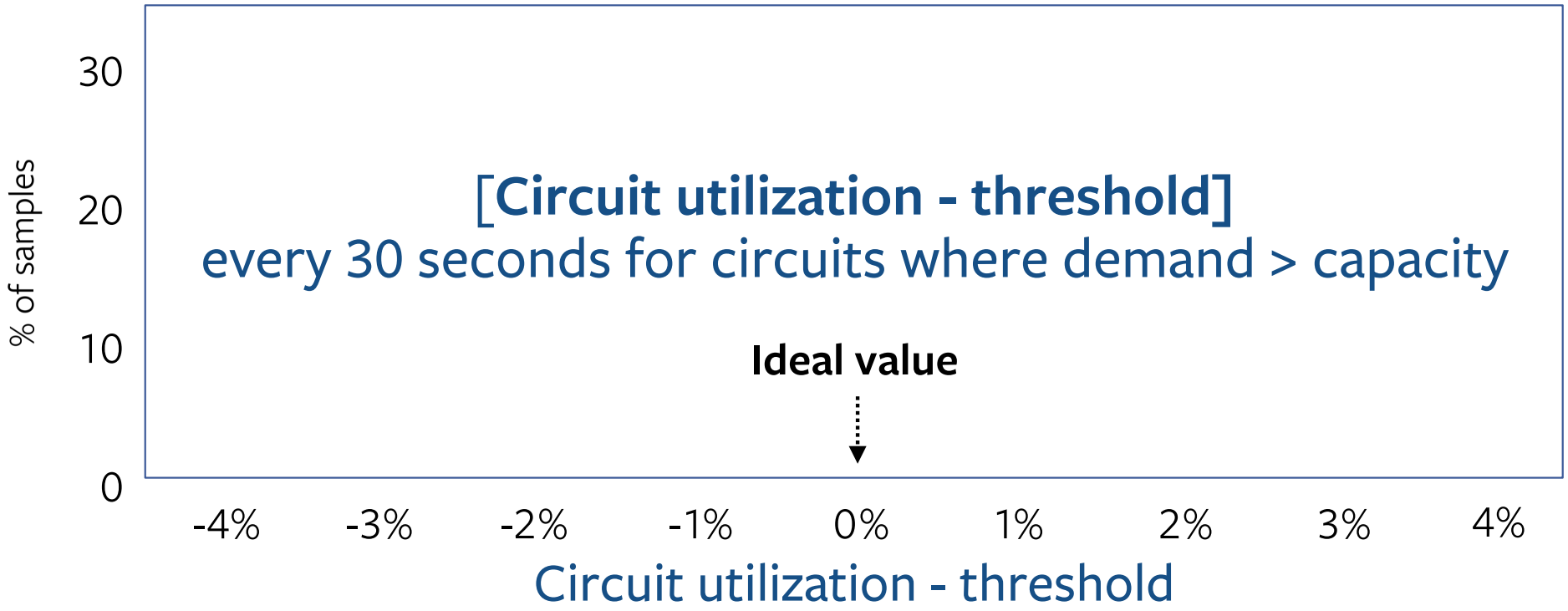
# Evaluating Congestion Avoidance

## Can we keep utilization at the threshold?



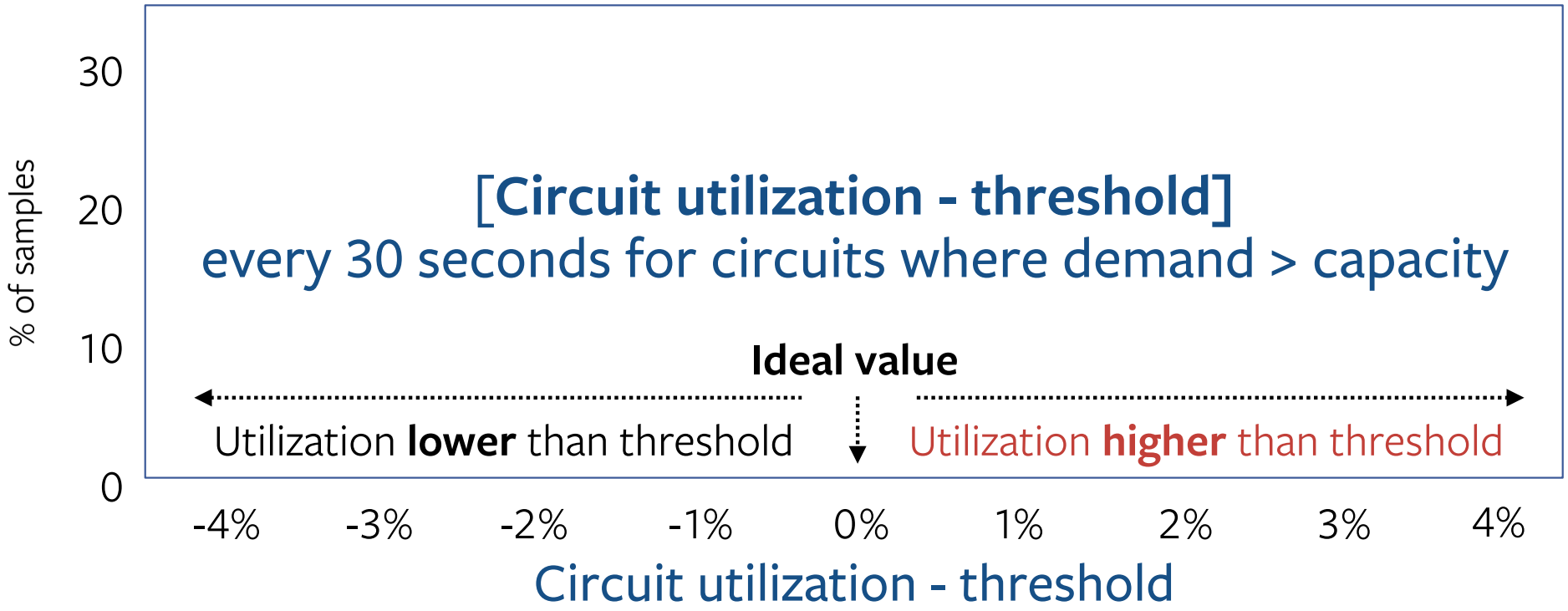
# Evaluating Congestion Avoidance

## Can we keep utilization at the threshold?



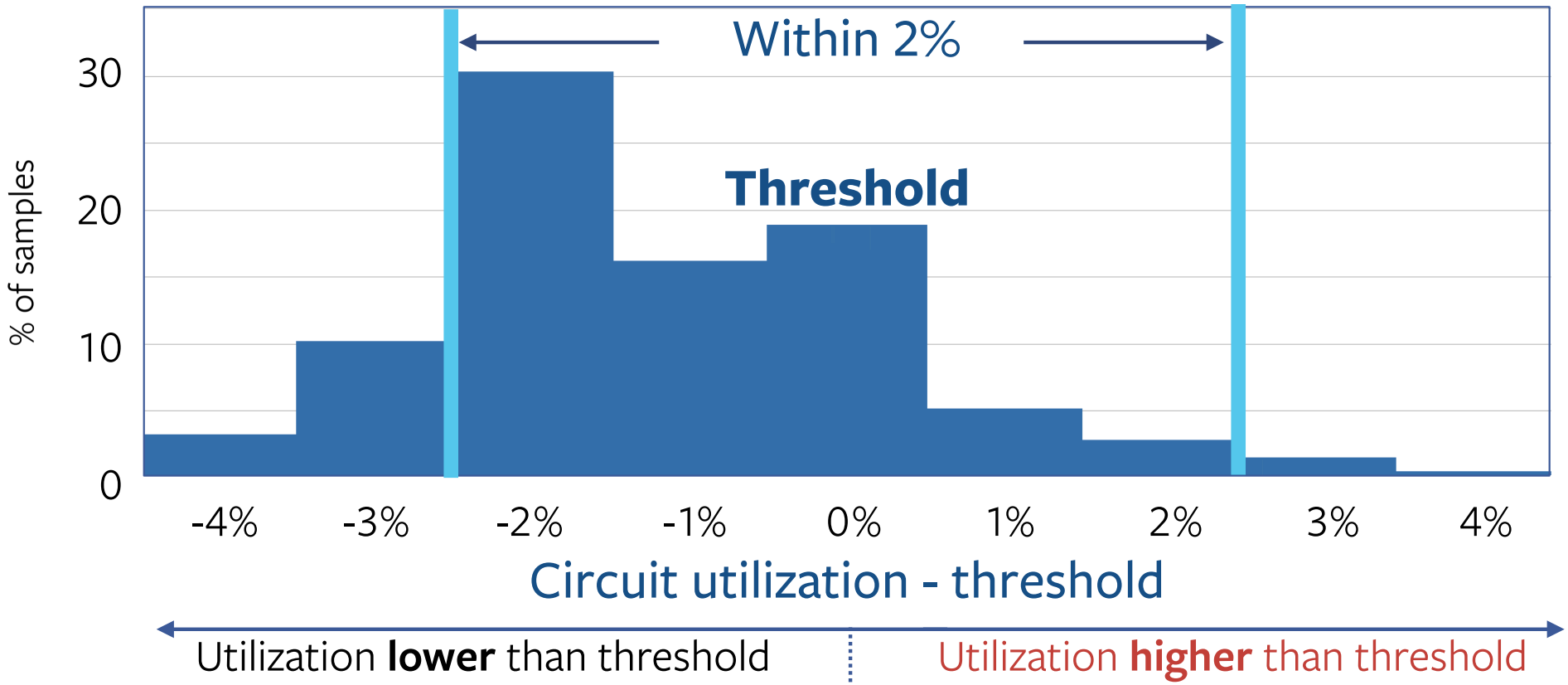
# Evaluating Congestion Avoidance

## Can we keep utilization at the threshold?



# Evaluating Congestion Avoidance

## Can we keep utilization at the threshold?



# Evaluating Congestion Avoidance

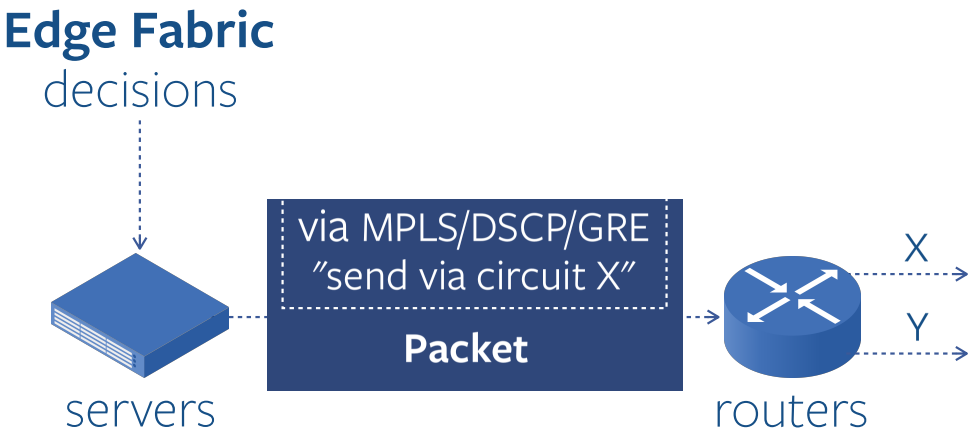
Does Edge Fabric prevent circuit congestion and packet drops? **Yes.**

Does Edge Fabric keep circuit utilization at prescribed threshold? **Yes.**

**Edge Fabric prevents packet loss while keeping circuit utilization high**

# Outline

# Evolution: Enacting Decisions




**V1:** Host-based routing

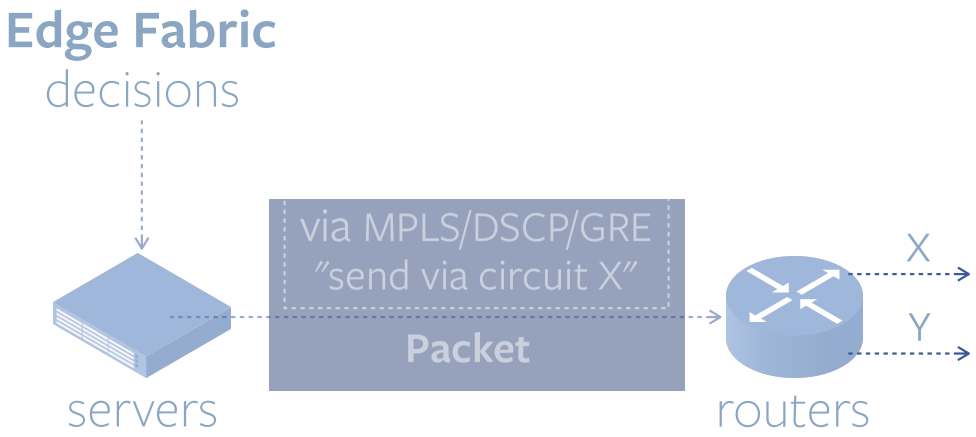
Overrides enacted by hosts

Hosts signal egress path per packet

**Key Challenge:** *Synchronization*

Routing state maintained across all hosts

# Evolution: Enacting Decisions




**V1:** Host-based routing

Overrides enacted by hosts

Hosts signal egress path per packet

**Key Challenge:** *Synchronization*

Routing state maintained across all hosts


**Today:** Edge-based routing

Overrides enacted by routers at edge

Hosts signal priority per packet

**No host synchronization required**

Flexible with DSCP signaling, fallback to BGP

# Evolution: Enacting Decisions

**Before:** Host-based routing

**Today:** Edge-based routing

Both provide the **capabilities** we want today

Preventing congestion, incorporating advanced policy, application-specific and performance-aware routing

# Evolution: Enacting Decisions

**Before:** Host-based routing

**Today:** Edge-based routing

Both provide the **capabilities** we want today

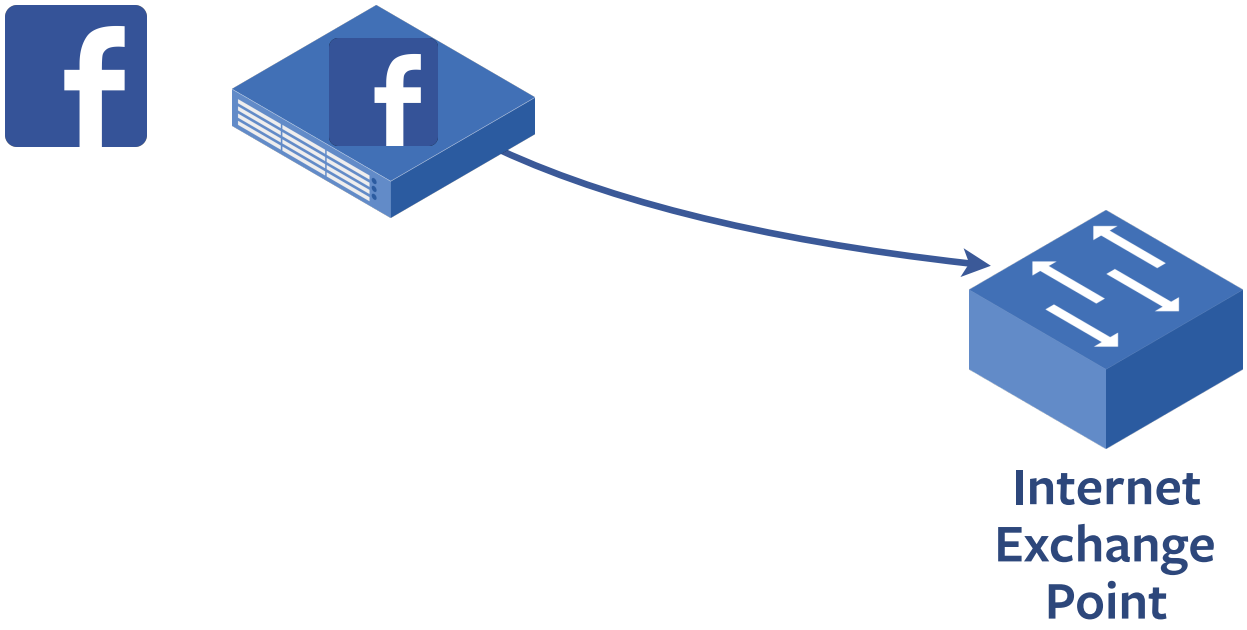
Preventing congestion, incorporating advanced policy, application-specific and performance-aware routing

**Edge-based** is best aligned with our design priorities

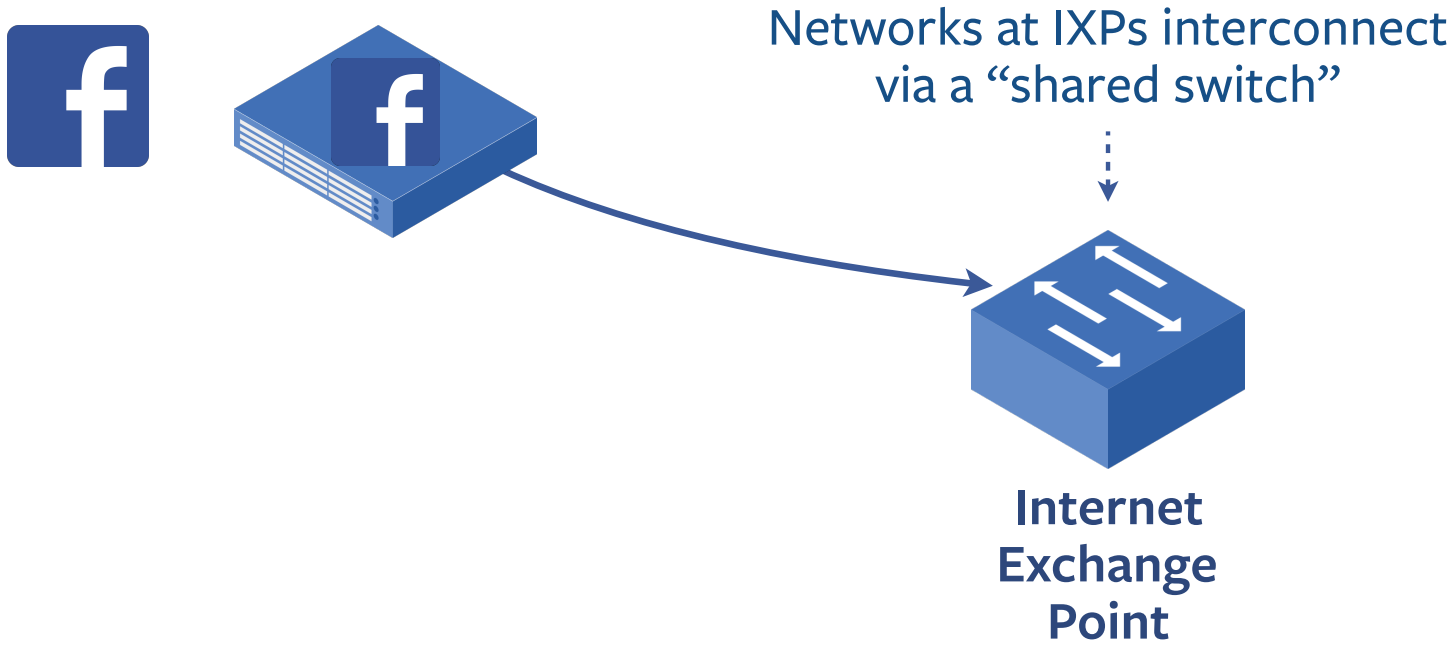
Operational simplicity

Ease of deployment

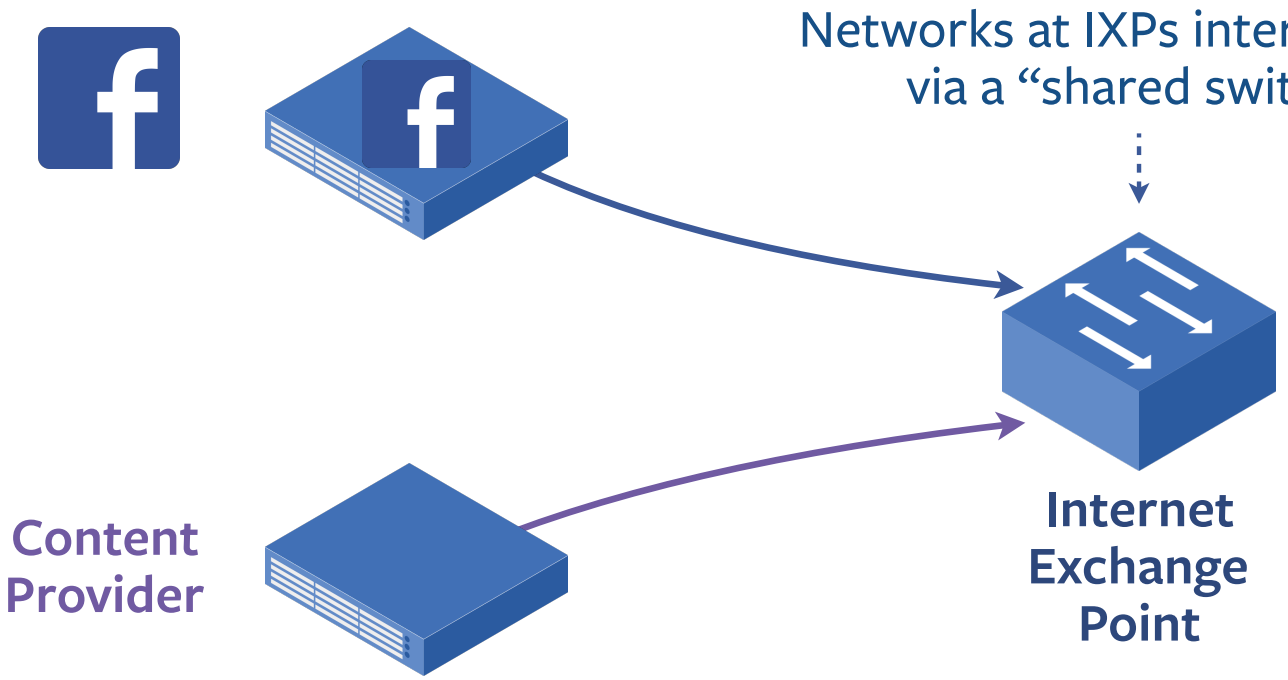
# Congestion Beyond Our Edge: IXPs



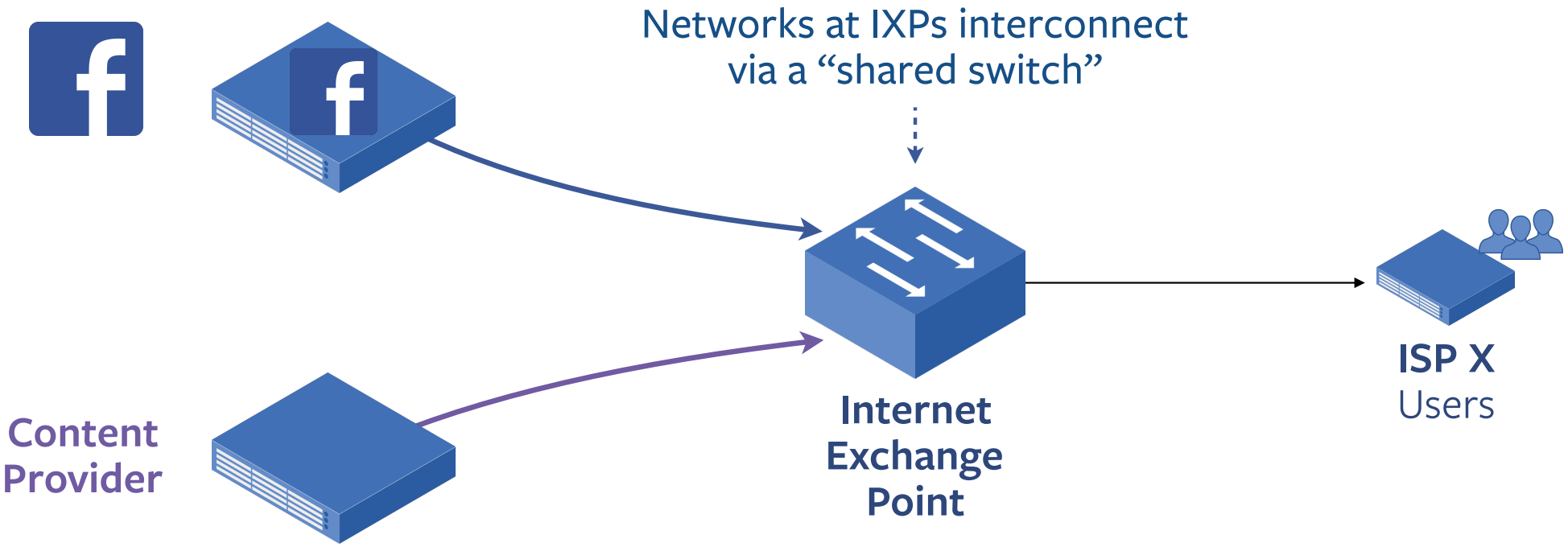
# Congestion Beyond Our Edge: IXPs



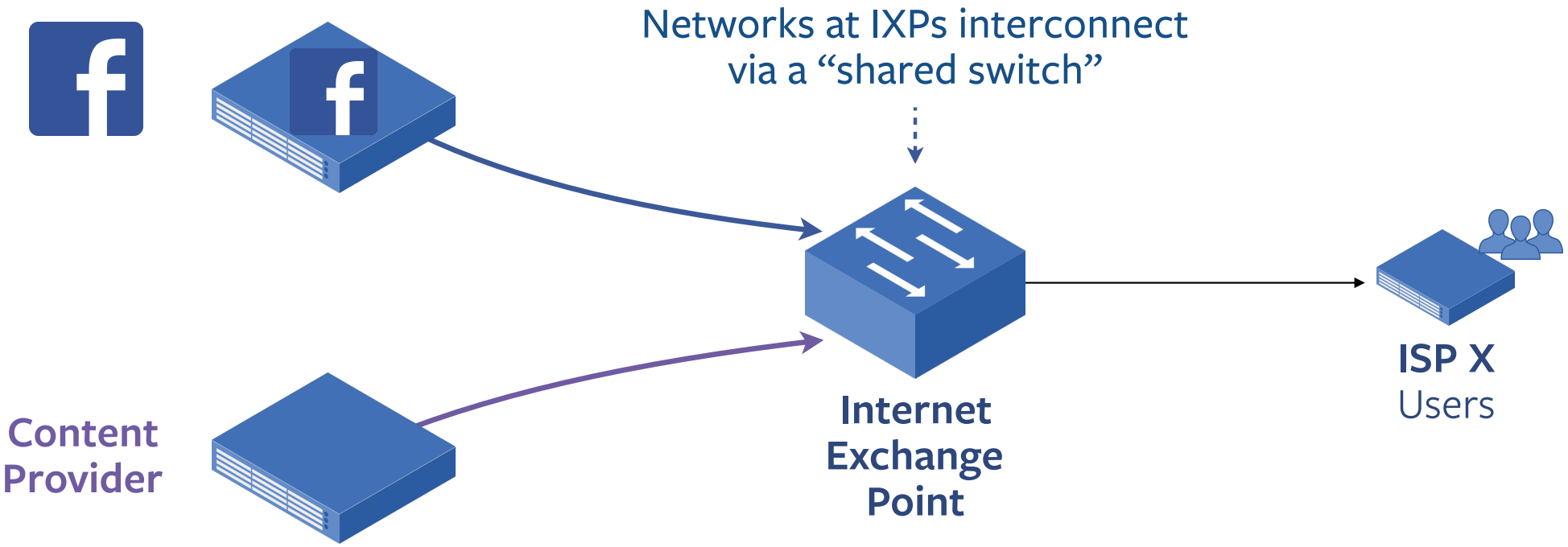
# Congestion Beyond Our Edge: IXPs



# Congestion Beyond Our Edge: IXPs

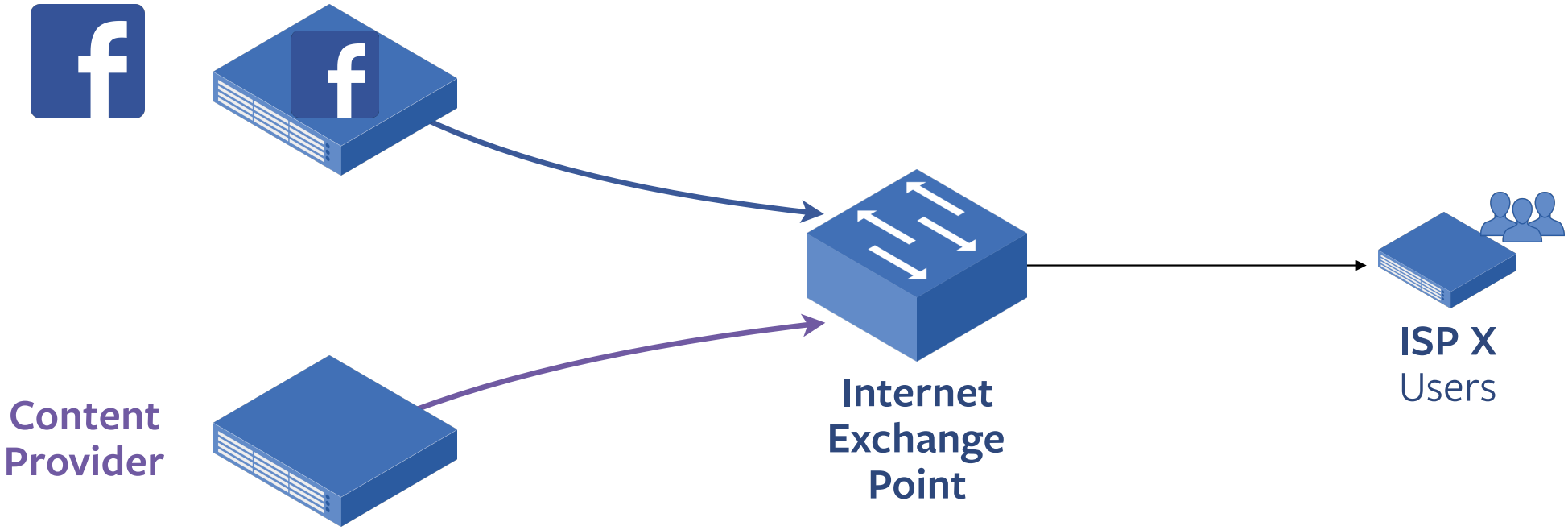


# Congestion Beyond Our Edge: IXPs

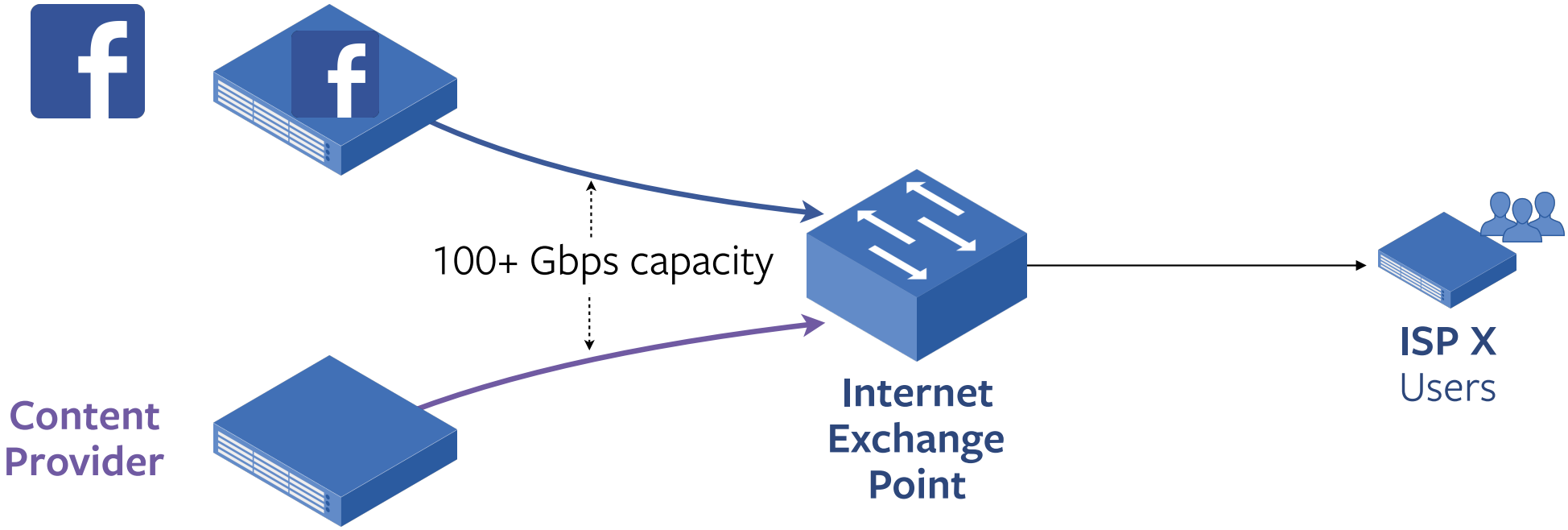



Internet Exchange Points remove barriers to interconnection

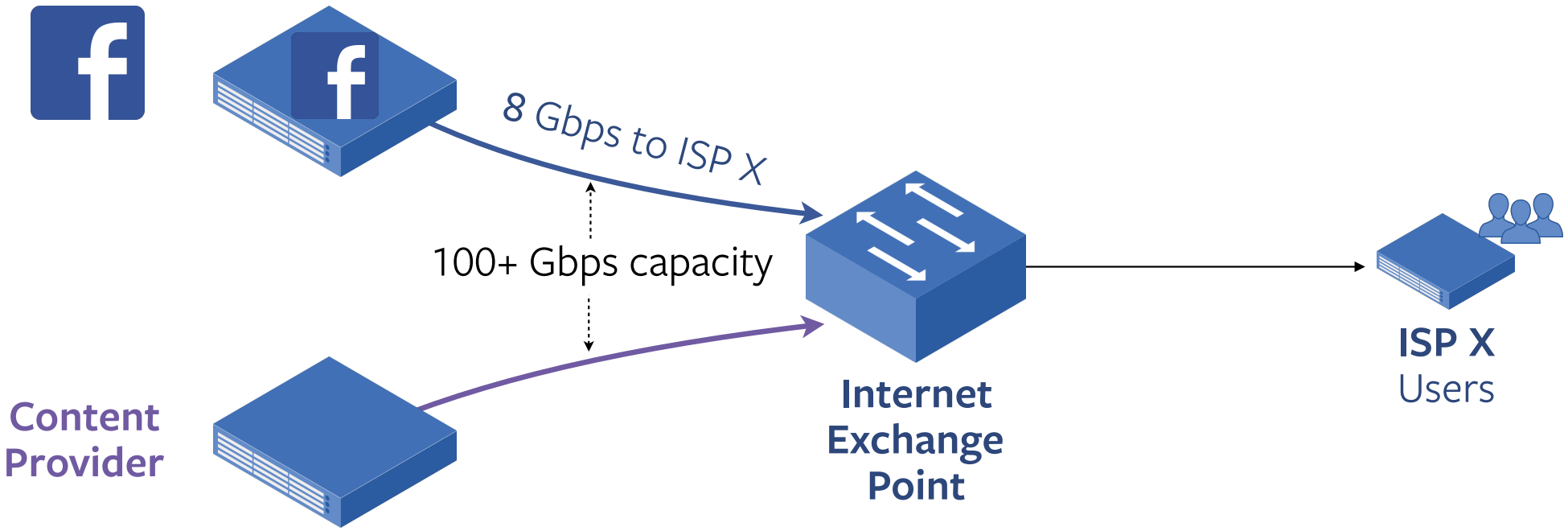
# Congestion Beyond Our Edge: IXPs



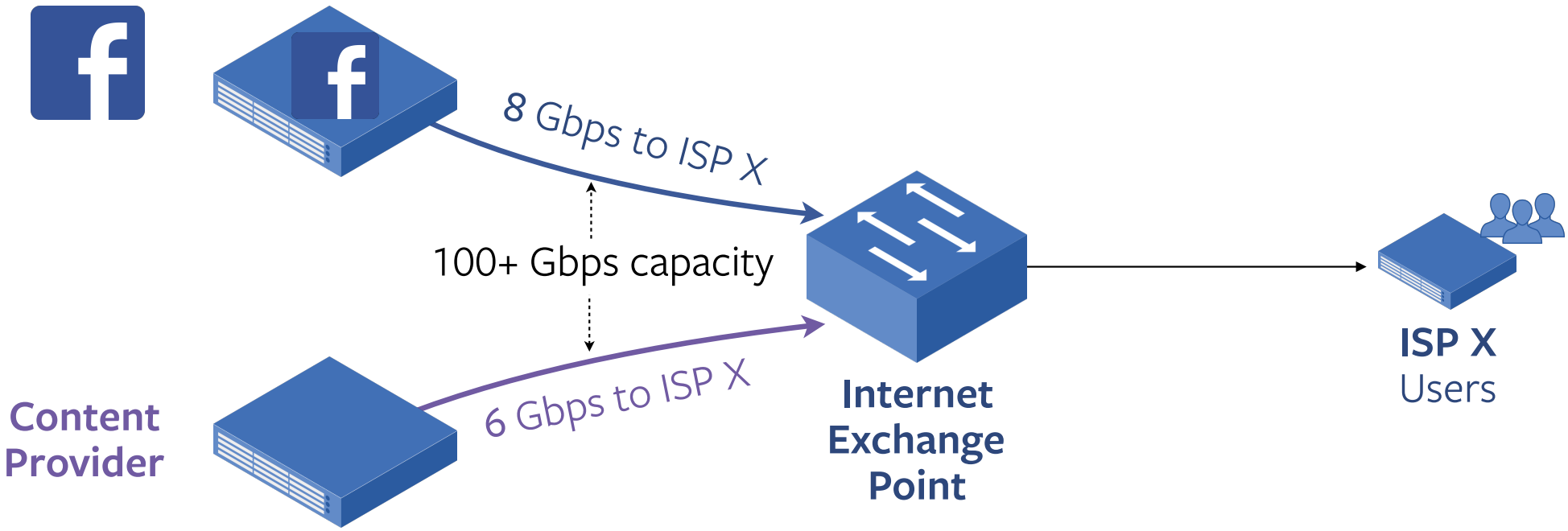
# Congestion Beyond Our Edge: IXPs



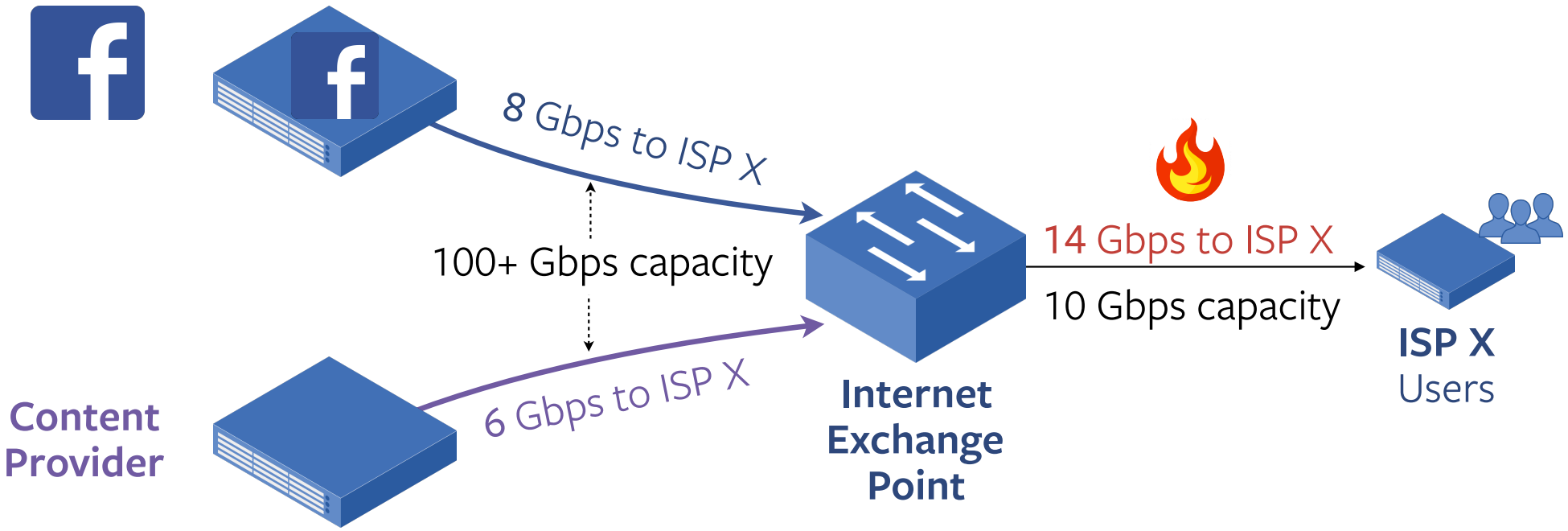
# Congestion Beyond Our Edge: IXPs



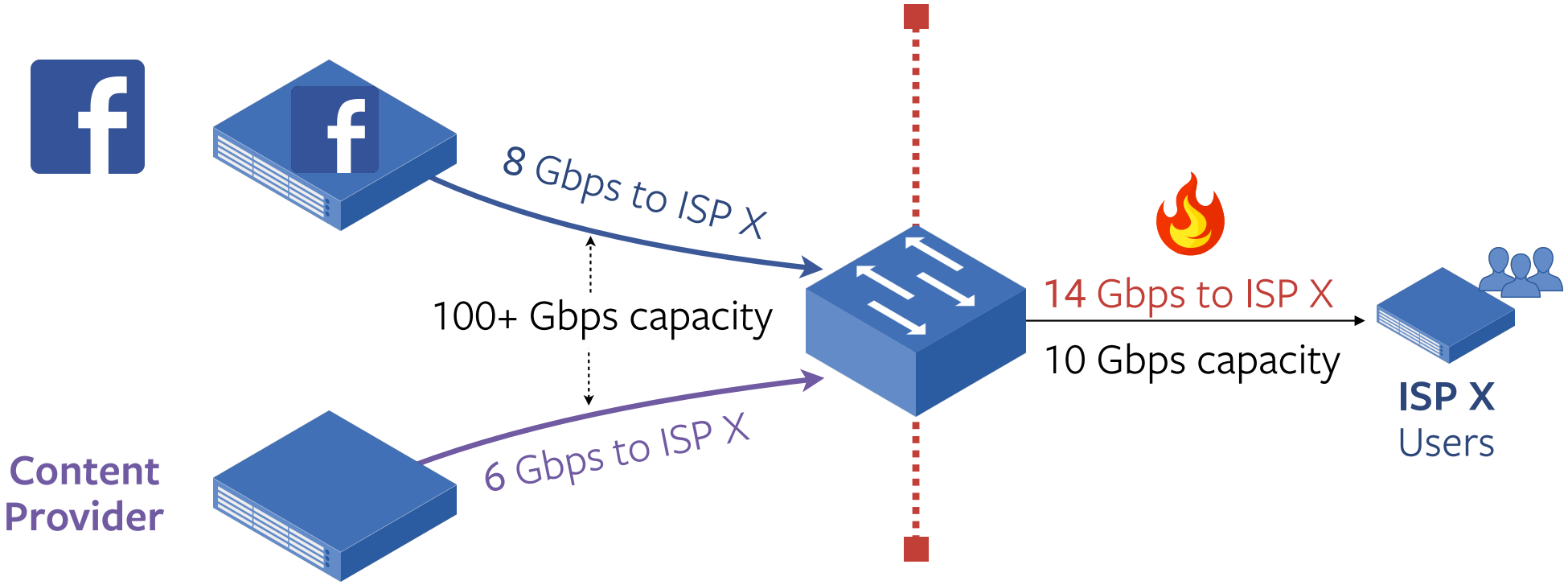
# Congestion Beyond Our Edge: IXPs



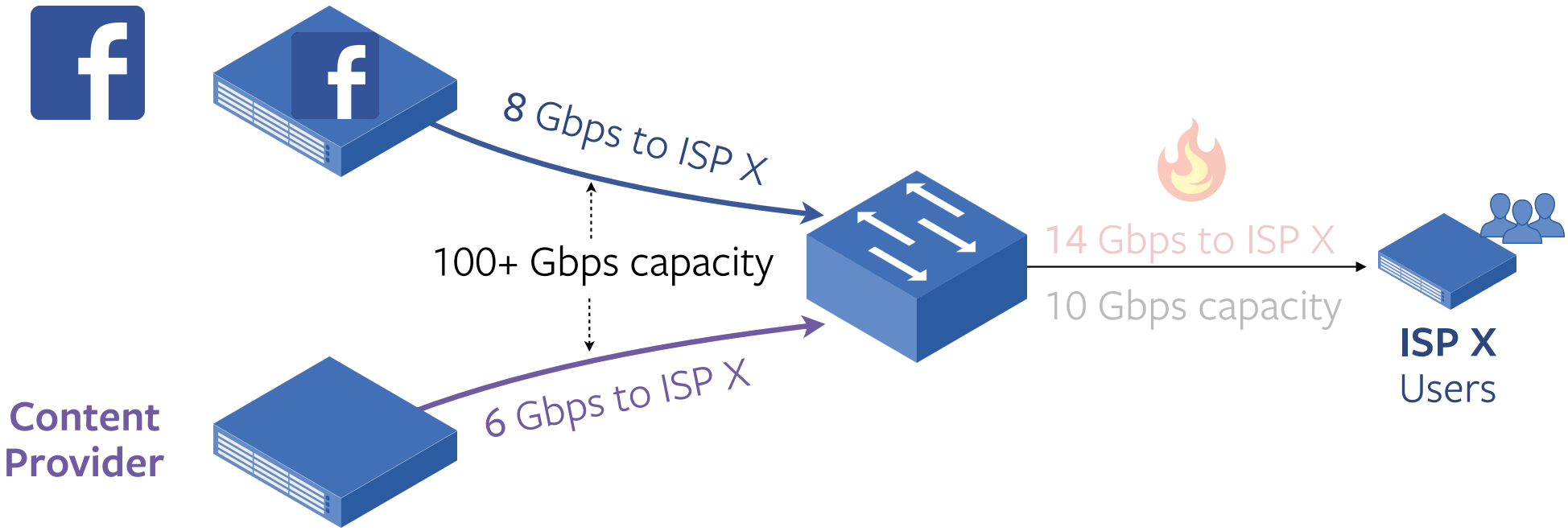
# Congestion Beyond Our Edge: IXPs



# Congestion Beyond Our Edge: IXPs

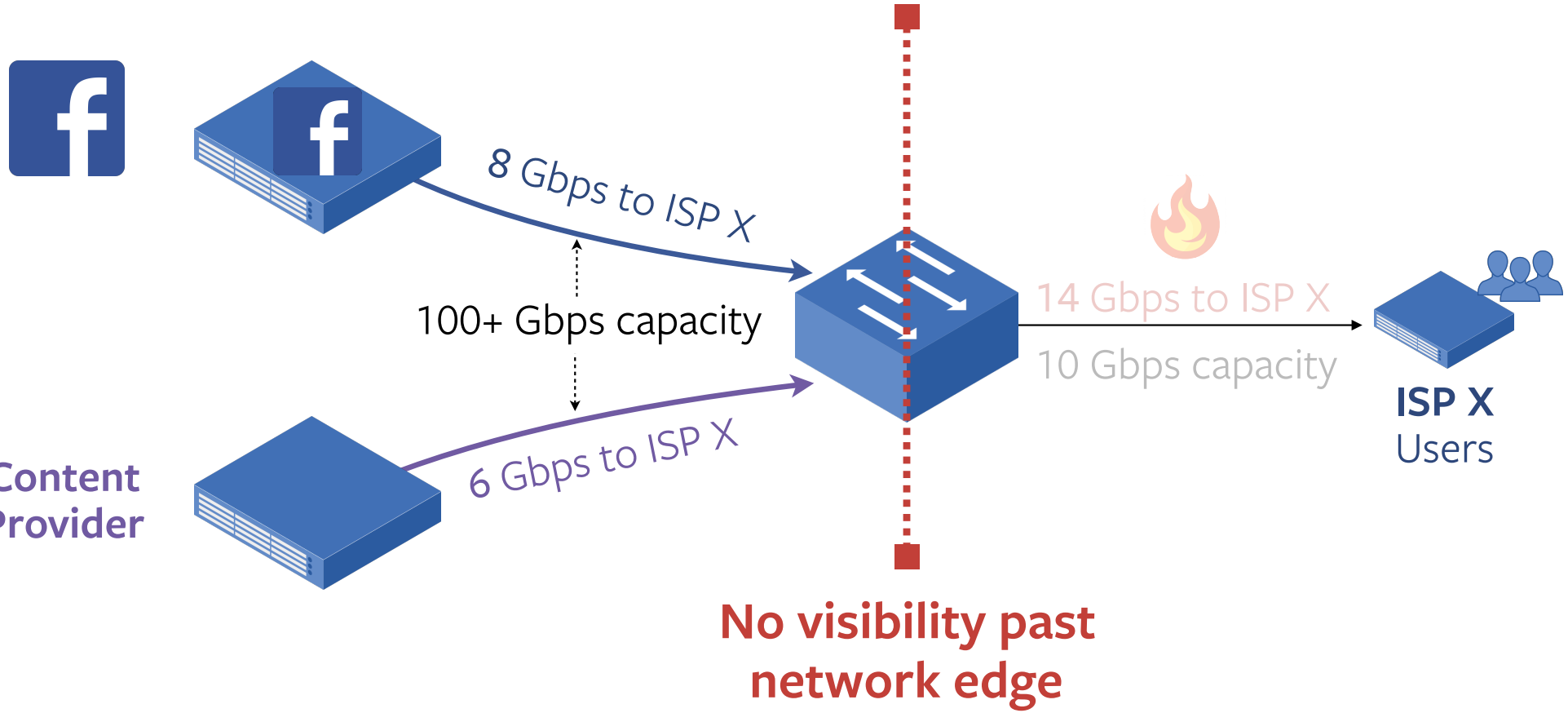


# Congestion Beyond Our Edge: IXPs

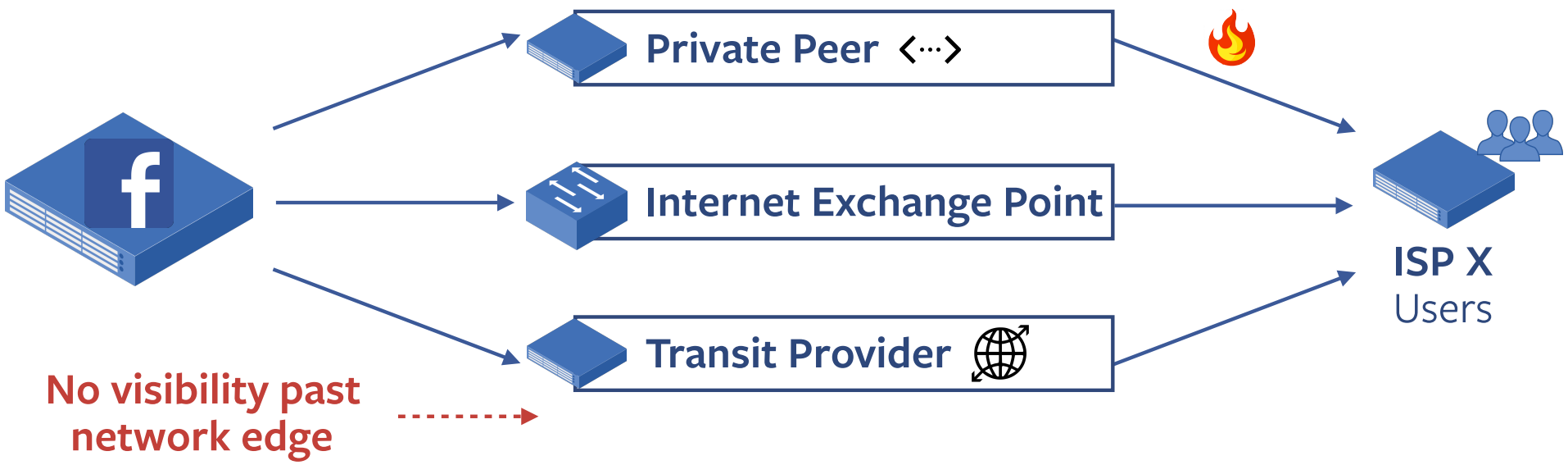



**No visibility past network edge**

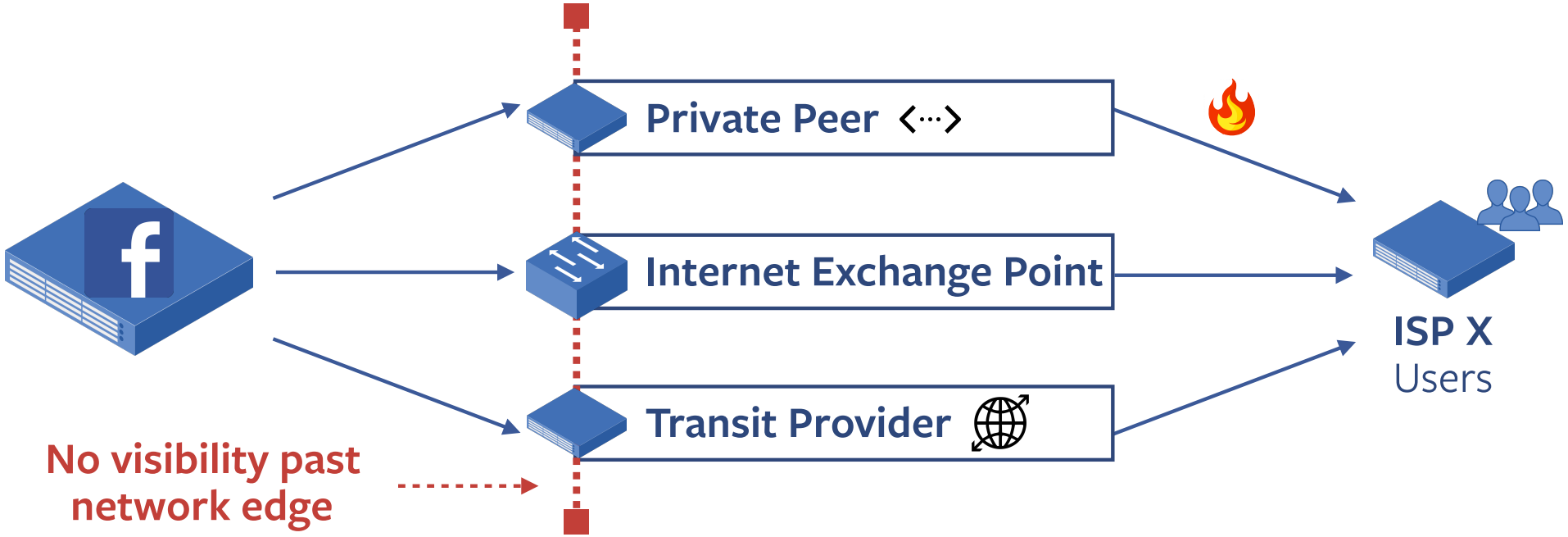
# Congestion Beyond Our Edge: IXPs



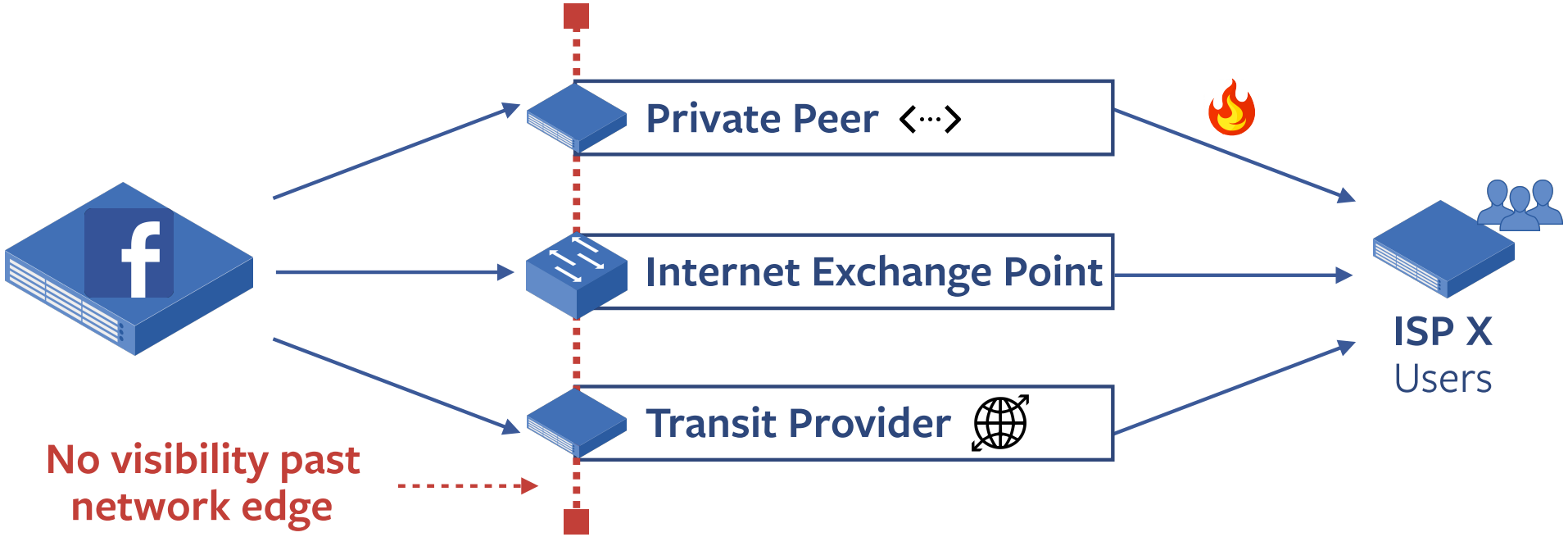
# Congestion Beyond Our Edge



# Congestion Beyond Our Edge



# Congestion Beyond Our Edge




Lack of visibility exists across interconnection types

# Identifying Congestion Beyond Our Edge

## Candidate Signals

# Identifying Congestion Beyond Our Edge

## Candidate Signals

**1 Gbps** Prefix traffic rates

# Identifying Congestion Beyond Our Edge

## Candidate Signals

~~1 Gbps~~ ~~Prefix traffic rates~~ Cross traffic beyond edge

# Identifying Congestion Beyond Our Edge

## Candidate Signals

~~1 Gbps~~ ~~Prefix traffic rates~~ Cross traffic beyond edge

**40 Gbps** Circuit capacities

# Identifying Congestion Beyond Our Edge

## Candidate Signals

1 ~~Gbps~~ ~~Prefix traffic rates~~ Cross traffic beyond edge

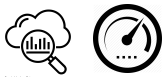
40 ~~Gbps~~ ~~Circuit capacities~~ Don’t know beyond edge

# Identifying Congestion Beyond Our Edge

## Candidate Signals

~~Prefix traffic rates~~ Cross traffic beyond edge

~~Circuit capacities~~ Don't know beyond edge



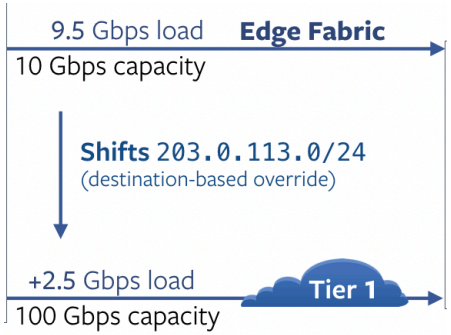
Route performance measurements → **Edge Fabric**

Must **infer** congestion from performance measurements

Example challenge: Latency increase due to **path change**, or **congestion**?

# Reacting to Congestion Beyond Our Edge

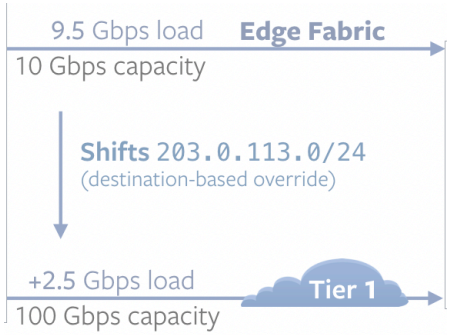
## How Much Traffic to Shift?




**Continuously probe** for capacity
**Discover** via trial and error

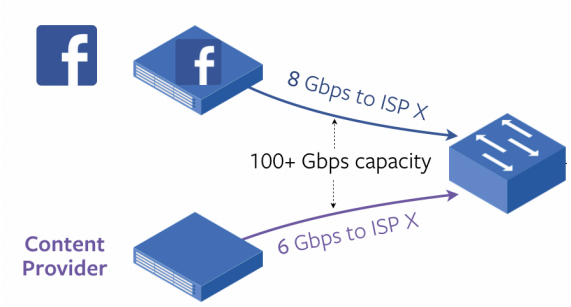
# Reacting to Congestion Beyond Our Edge

## How Much Traffic to Shift?




**Continuously probe** for capacity
**Discover** via trial and error

## Interactions with Other Networks




**Others may respond** to congestion signals
**Oscillations** between networks

# What’s New?

# What's New?

## Problem has been around for a decade.

BGP does not consider demand, capacity or performance

# What's New?

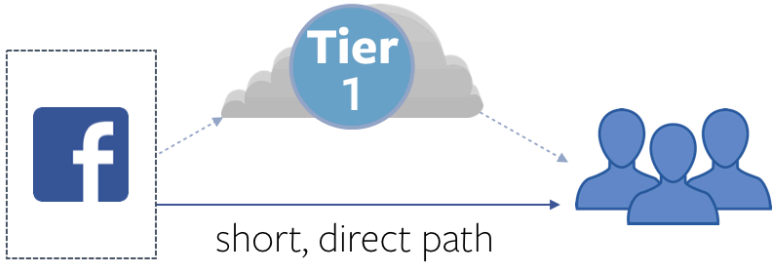
**Problem has been around for a decade.**

BGP does not consider demand, capacity or performance

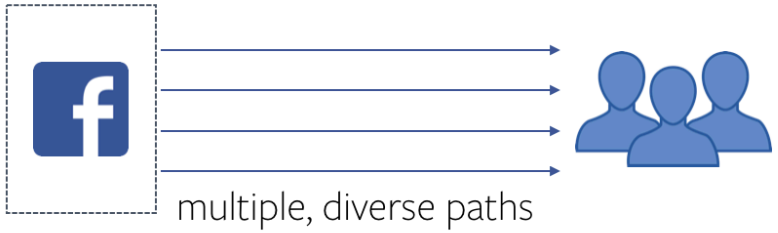
**Scale of connectivity, traffic, and QoS demands brings new challenges and opportunities**

# Conclusion

## Benefits of Rich Interconnection




short, direct paths
bypass transit providers




substantial path diversity

# Conclusion

| | |
|---|---|
| objective | deliver traffic with the best performance possible |

# Conclusion

| | |
|---|---|
| objective | deliver traffic with the best performance possible |
| challenge | BGP does not consider demand, capacity or performance |

# Conclusion

| | |
|---|---|
| objective | deliver traffic with the best performance possible |
| challenge | BGP does not consider demand, capacity or performance |

With **Edge Fabric**, we sidestep BGP's limitations by **shifting control from routers to software**

| | |
|---|---|
| **result** | more efficient network, better performance for our users |

# Backup Slides

# **Edge Fabric** and Google's **Espresso**

## **Both systems**

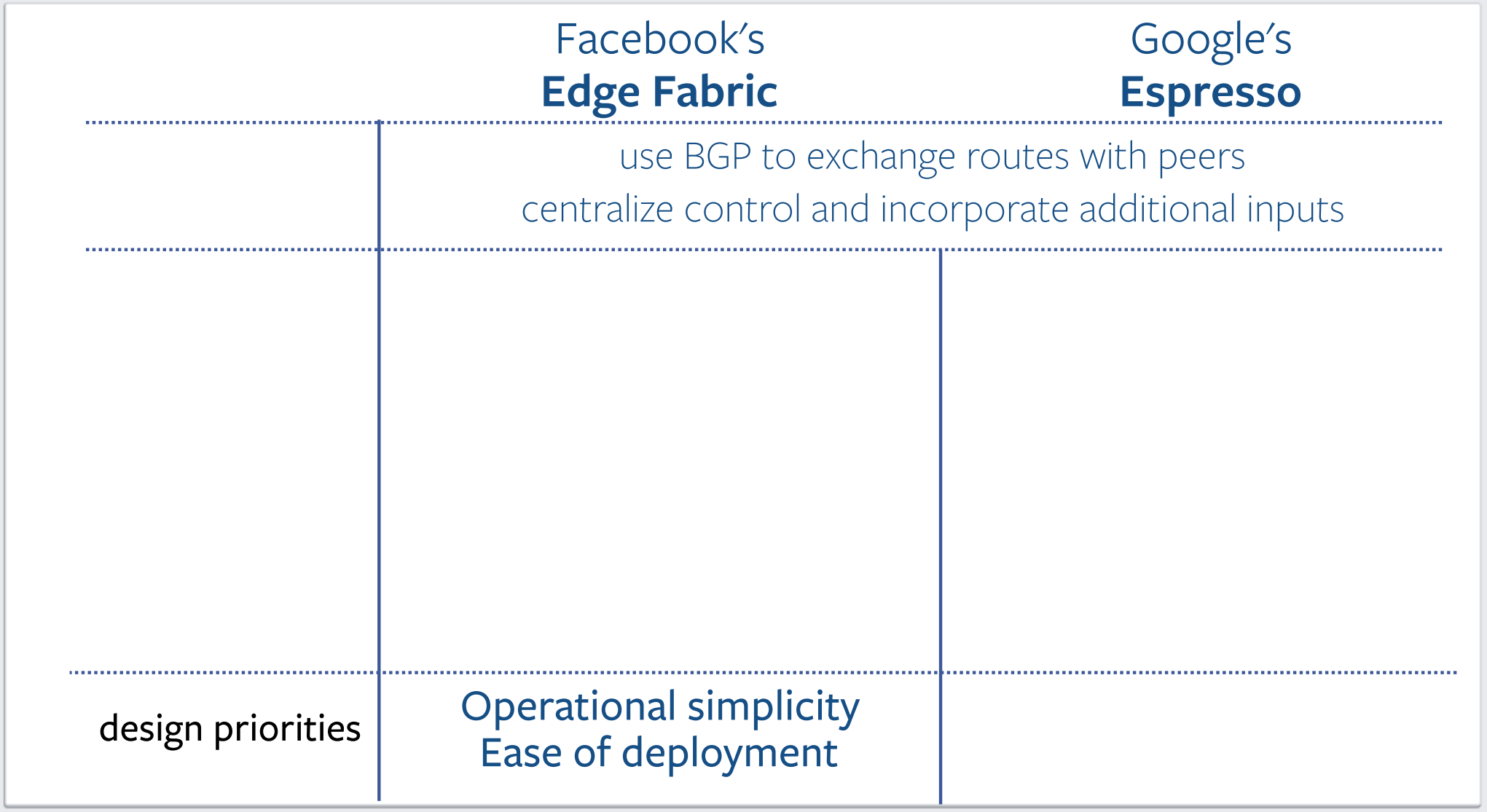
use BGP to exchange routes with peers

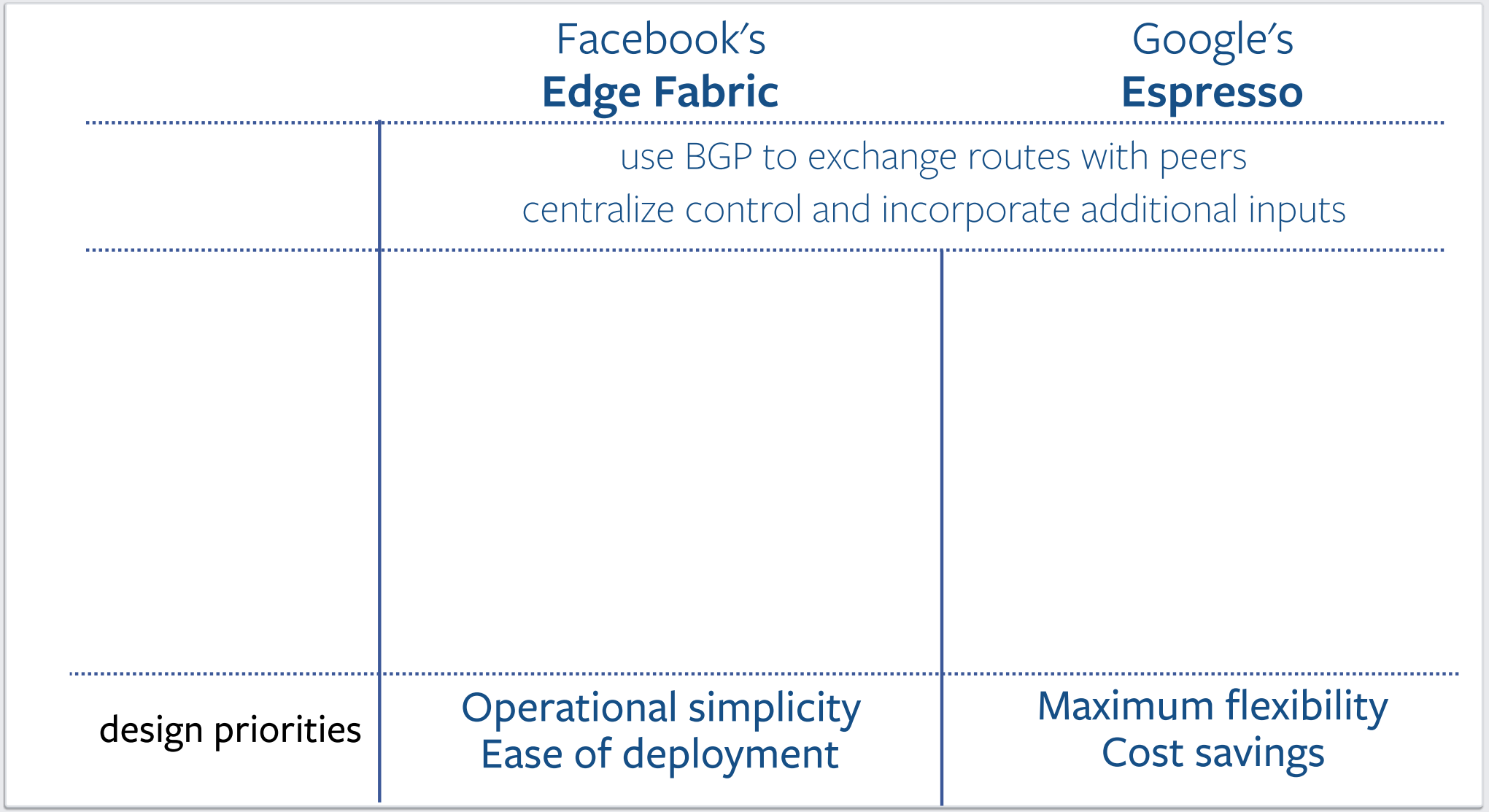
# **Edge Fabric** and Google's **Espresso**

**Both systems**

use BGP to exchange routes with peers

focus on centralizing control and incorporating additional inputs


Facebook's
Edge Fabric
Google's
Espresso
use BGP to exchange routes with peers
centralize control and incorporate additional inputs
design priorities
Operational simplicity
Ease of deployment


Facebook's
Edge Fabric
Google's
Espresso
use BGP to exchange routes with peers
centralize control and incorporate additional inputs
design priorities
Operational simplicity
Ease of deployment
Maximum flexibility
Cost savings

| | Facebook's **Edge Fabric** | Google's **Espresso** |
|---|---|---|
| | use BGP to exchange routes with peers<br>centralize control and incorporate additional inputs | |
| **edge device** | **router** | **MPLS switch** |
| enacts decisions via | | |
| role of hosts | | |
| decision granularity | | |
| routing options | | |
| **design priorities** | Operational simplicity<br>Ease of deployment | Maximum flexibility<br>Cost savings |

| | Facebook's **Edge Fabric** | Google's **Espresso** |
|---|---|---|
| | use BGP to exchange routes with peers<br>centralize control and incorporate additional inputs | |
| edge device | router | MPLS switch |
| **enacts decisions via** | **BGP injections to routers** | **host-based overrides** |
| role of hosts | | |
| decision granularity | | |
| routing options | | |
| **design priorities** | Operational simplicity<br>Ease of deployment | Maximum flexibility<br>Cost savings |

| | Facebook's **Edge Fabric** | Google's **Espresso** |
|---|---|---|
| | use BGP to exchange routes with peers<br>centralize control and incorporate additional inputs | |
| edge device | router | MPLS switch |
| enacts decisions via | BGP injections to routers | host-based overrides |
| **role of hosts** | **mark packet's priority** | **select packet's route** |
| decision granularity | | |
| routing options | | |
| **design priorities** | Operational simplicity<br>Ease of deployment | Maximum flexibility<br>Cost savings |

| | Facebook's **Edge Fabric** | Google's **Espresso** |
|---|---|---|
| | use BGP to exchange routes with peers<br>centralize control and incorporate additional inputs | |
| edge device | router | MPLS switch |
| enacts decisions via | BGP injections to routers | host-based overrides |
| role of hosts | mark packet's priority | select packet's route |
| **decision granularity** | **<destination, priority/class>** | **packet** |
| routing options | | |
| **design priorities** | Operational simplicity<br>Ease of deployment | Maximum flexibility<br>Cost savings |

| | Facebook's **Edge Fabric** | Google's **Espresso** |
|---|---|---|
| | use BGP to exchange routes with peers<br>centralize control and incorporate additional inputs | |
| edge device | router | MPLS switch |
| enacts decisions via | BGP injections to routers | host-based overrides |
| role of hosts | mark packet's priority | select packet's route |
| decision granularity | <destination, priority/class> | packet |
| routing options | **per-PoP** | **global** |
| **design priorities** | Operational simplicity<br>Ease of deployment | Maximum flexibility<br>Cost savings |