Methods and Techniques for Disruption-free Network Reconfiguration

Laurent Vanbever

PhD defense

Advisor: Olivier Bonaventure

October 4, 2012

Methods and Techniques for Disruption-free Network Reconfiguration

1 **Background**

What is a network?

2 **Intradomain reconfiguration**

Find a reconfiguration ordering

3 **Interdomain reconfiguration**

Overcome inherent complexity

Methods and Techniques for Disruption-free Network Reconfiguration

1 Background

What is a network?

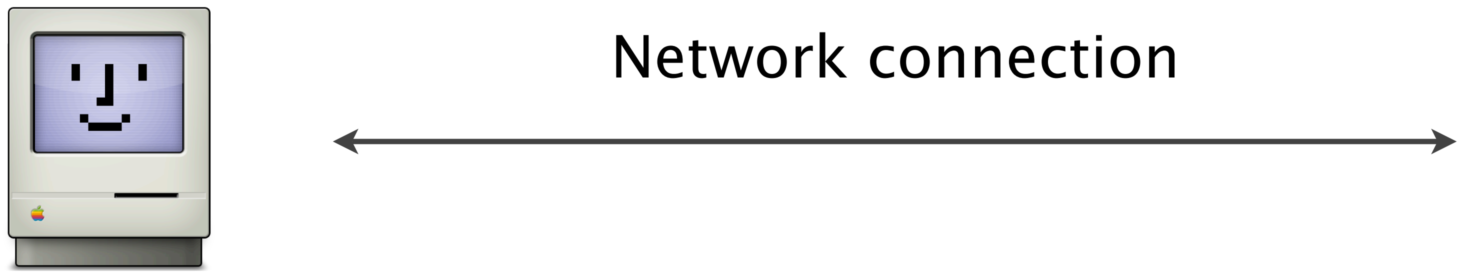
Intradomain reconfiguration

Find a reconfiguration ordering

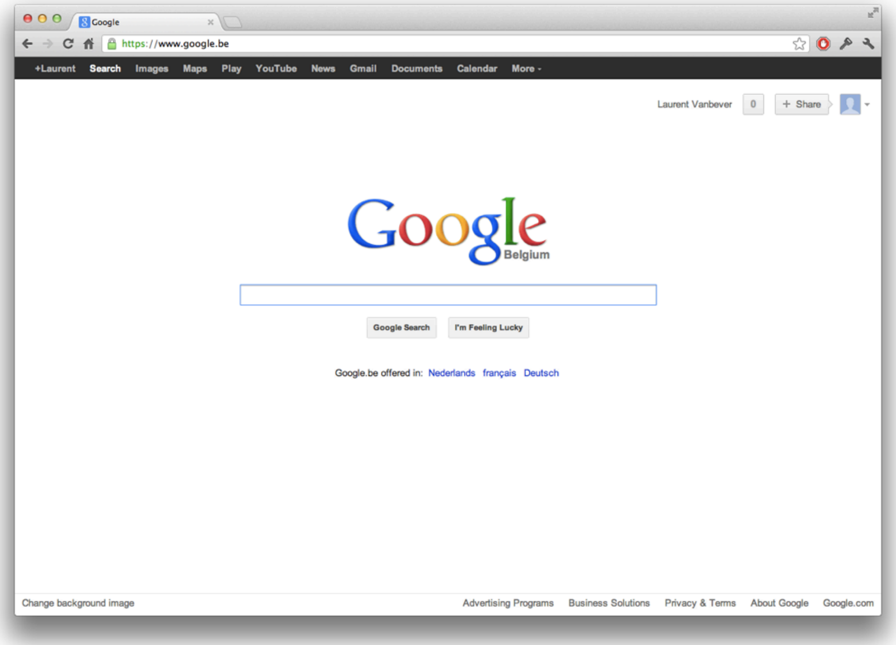
Interdomain reconfiguration

Overcome inherent complexity

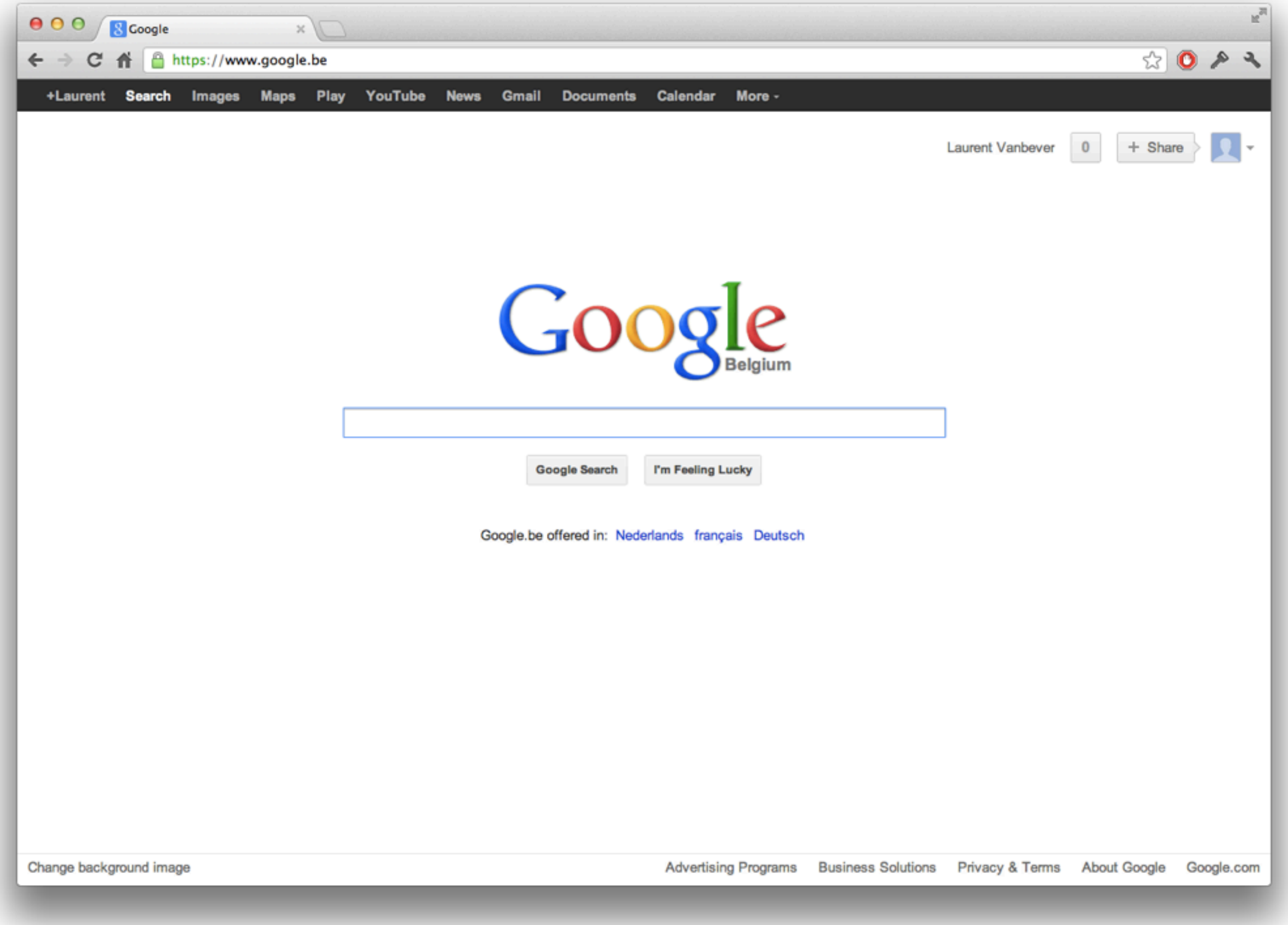
For a lot of people, this is what the Internet looks like



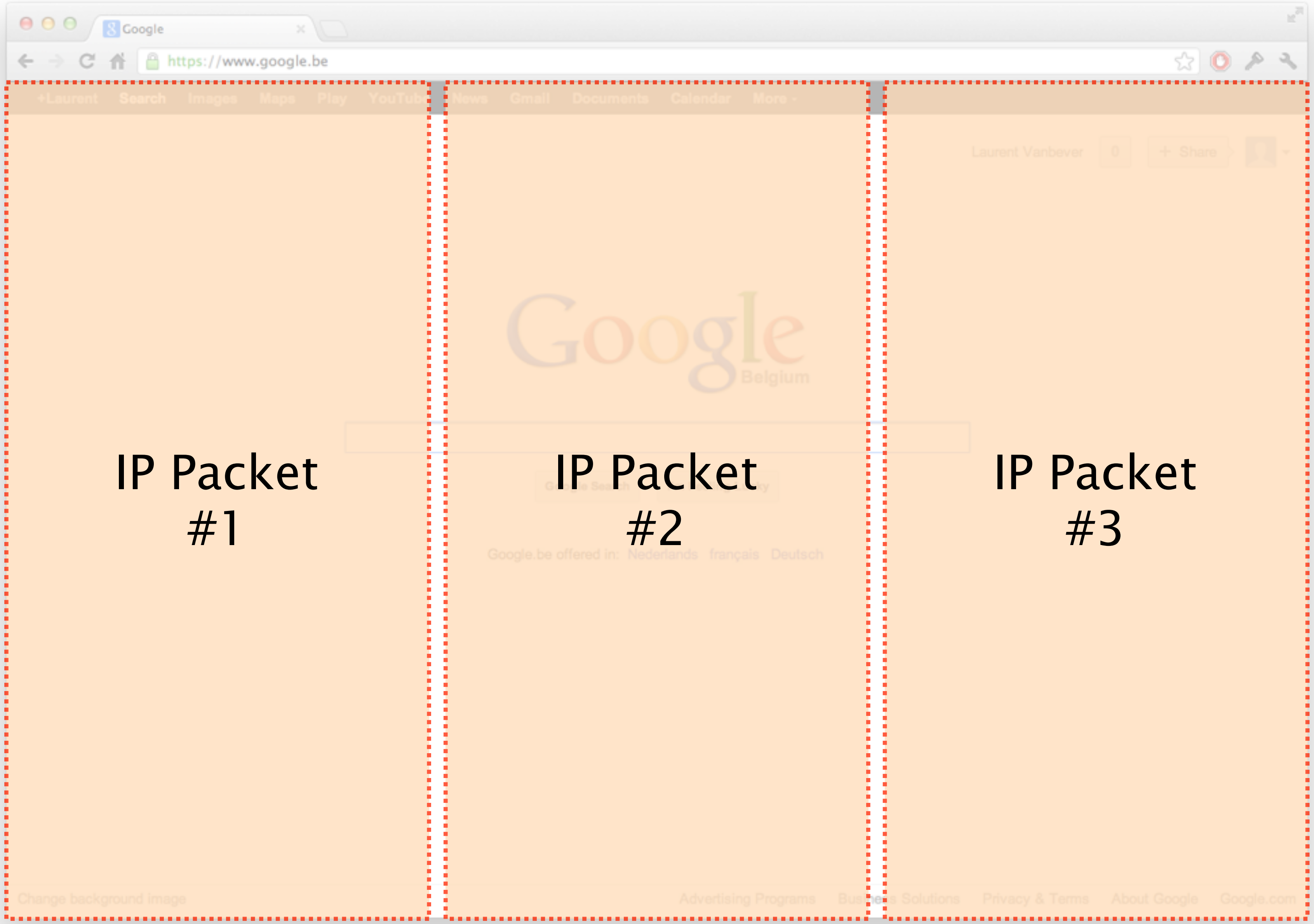
Network connection



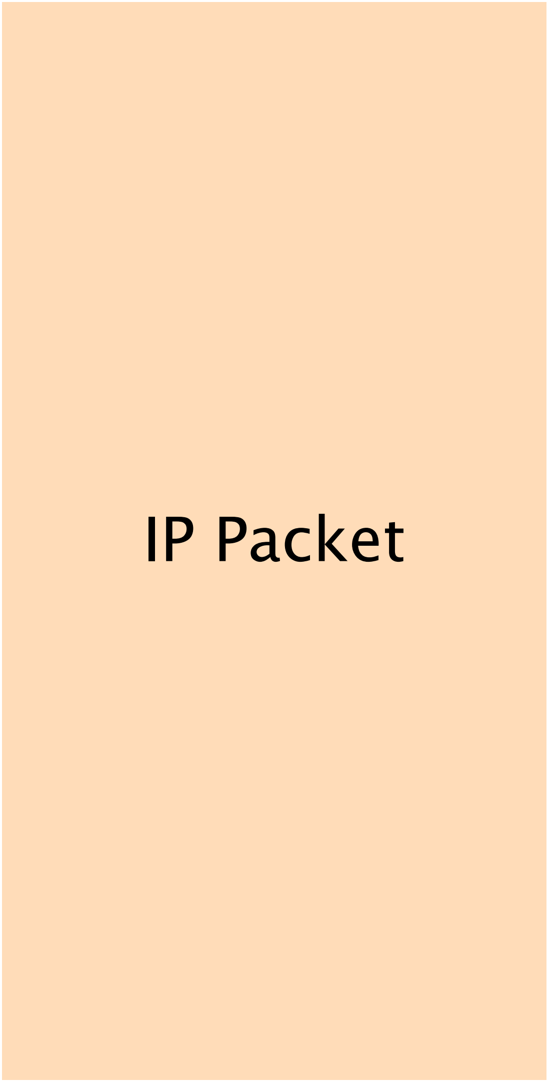
Data exchanged over the Internet are fragmented into small chuncks



Data exchanged over the Internet are fragmented into small chuncks: IP packets

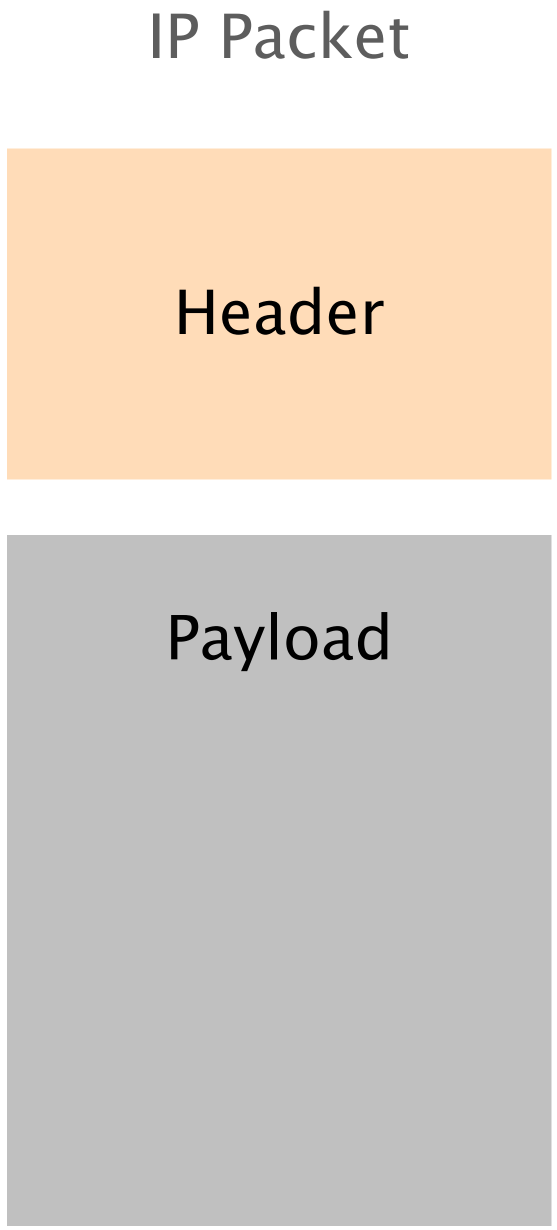


Data exchanged over the Internet are fragmented into small chuncks: IP packets



IP Packet

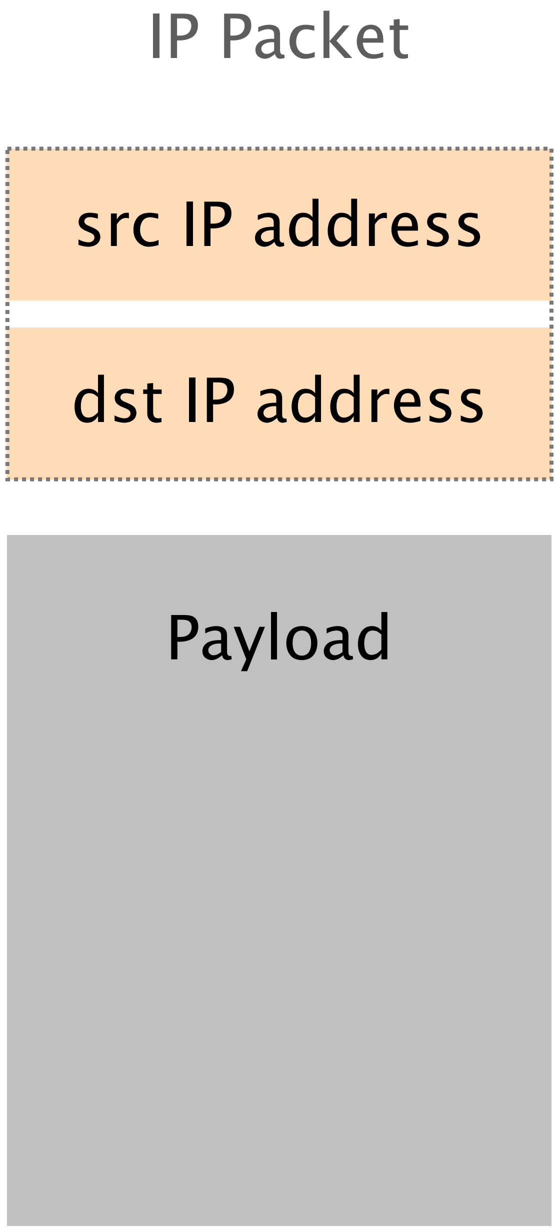
An IP packet is composed of two parts: the header and the payload



Metadata used to forward traffic

Content received by the application

An IP packet is composed of two parts: the header and the payload



IP Packet

Identify the source end-host

src IP address

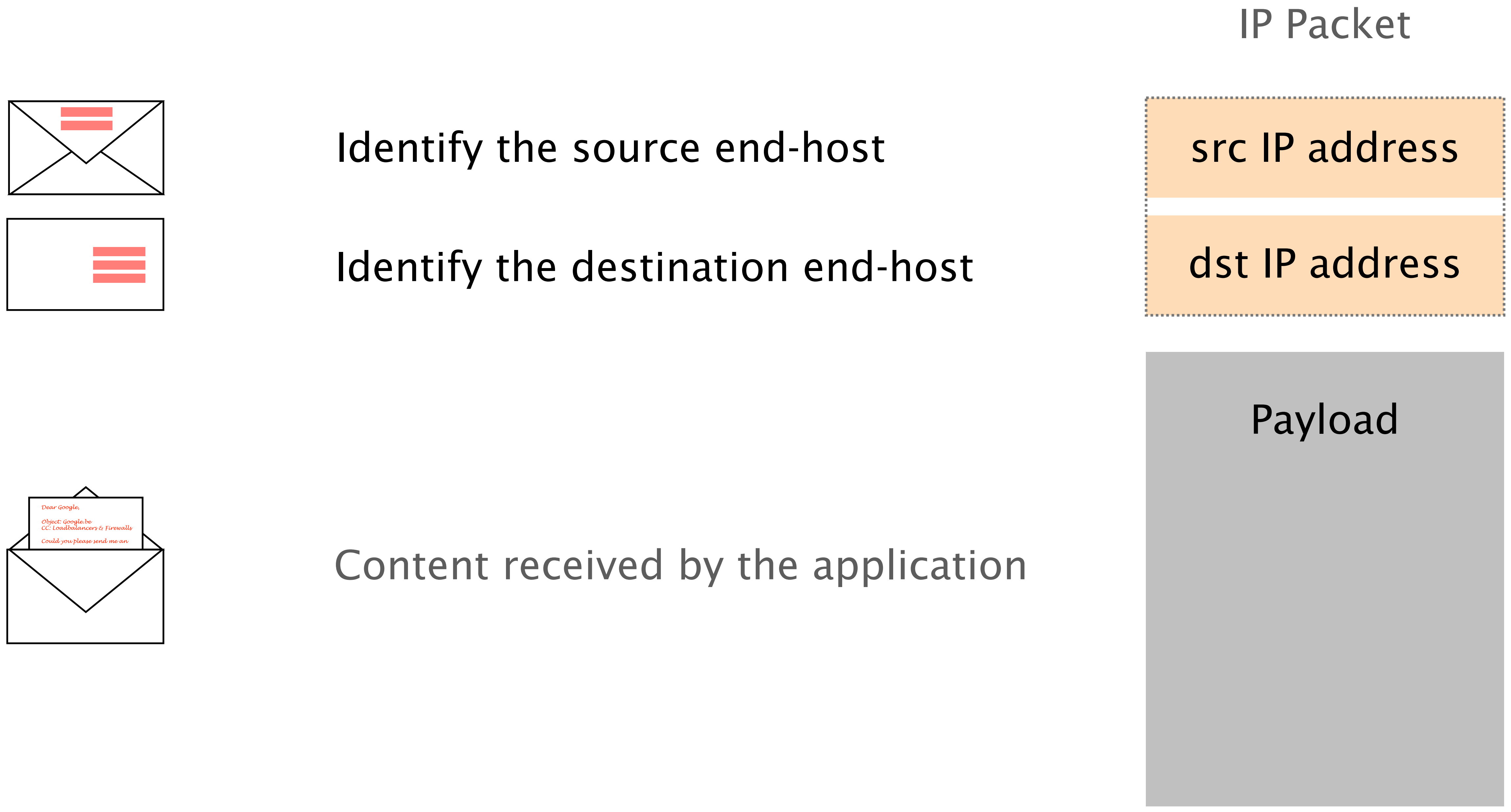
Identify the destination end-host

dst IP address

Payload

Content received by the application

An IP packet is composed of two parts: the header and the payload



IP Packet

Identify the source end-host

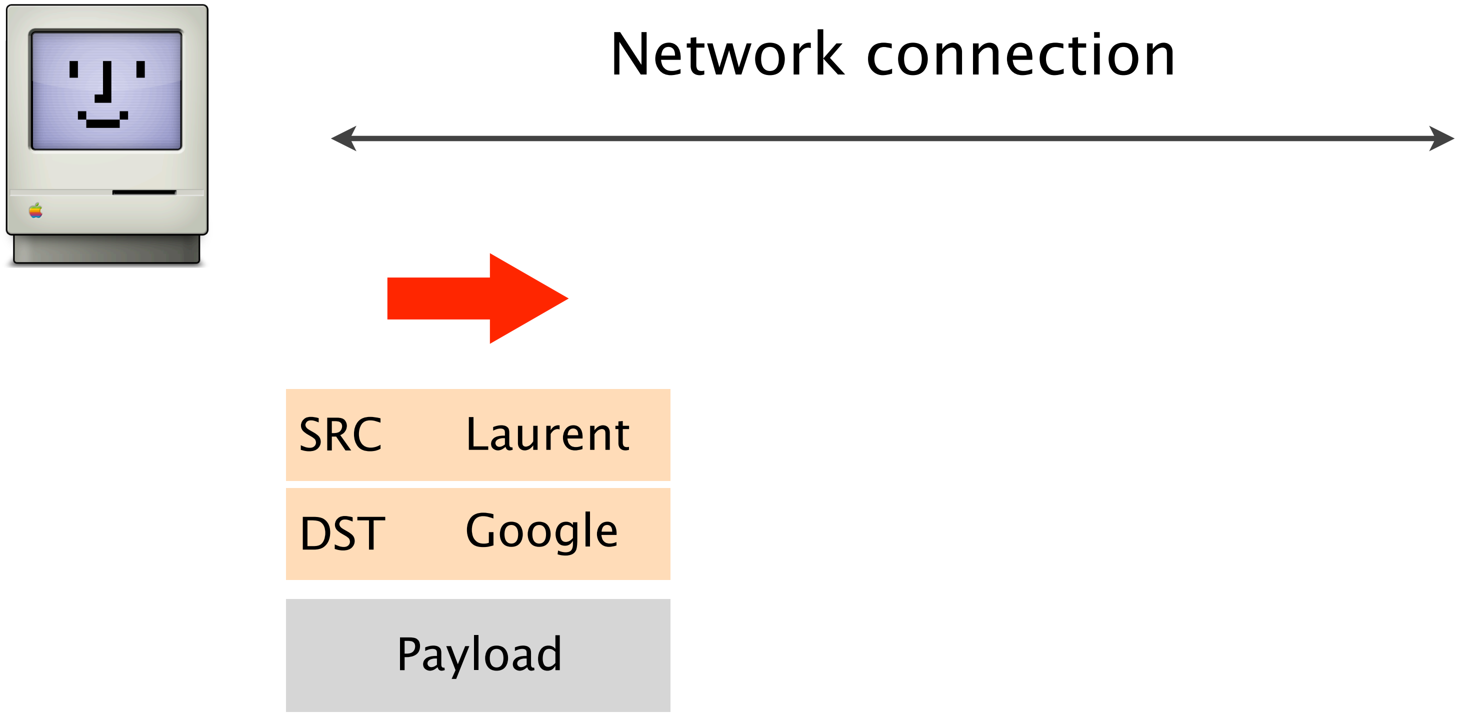
src IP address

Identify the destination end-host

dst IP address

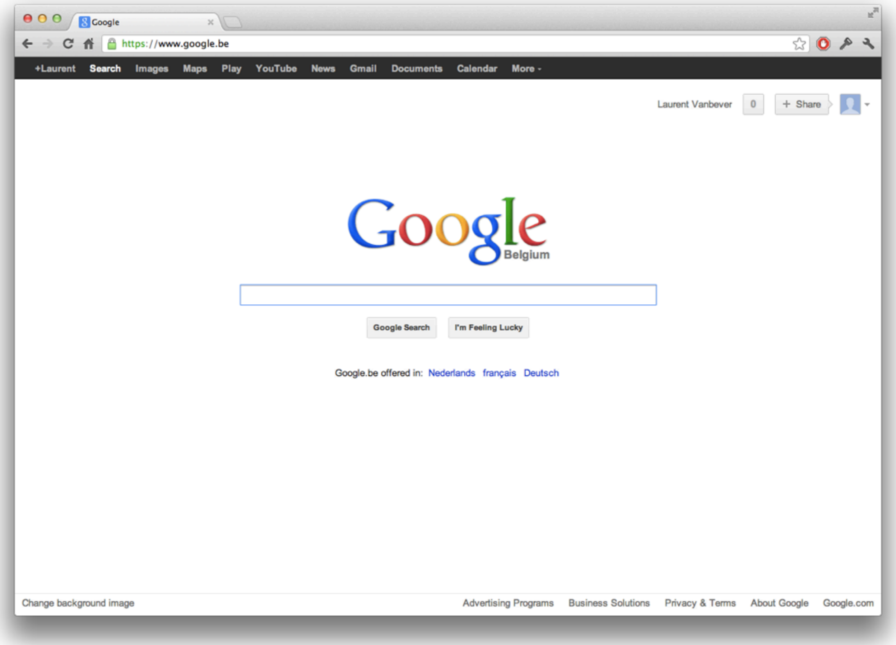
Payload

Content received by the application



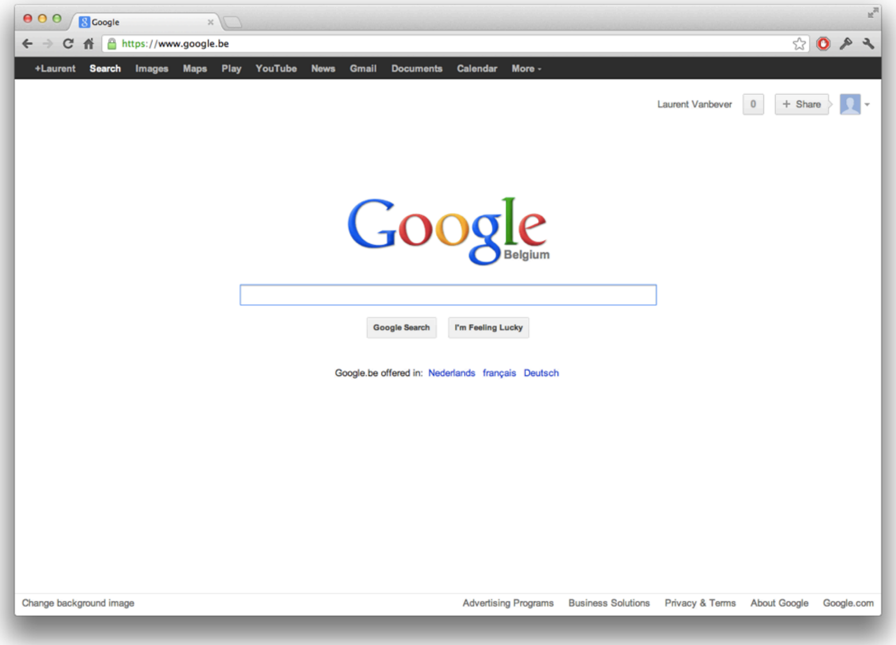
Network connection

SRC	Laurent
DST	Google
Payload	



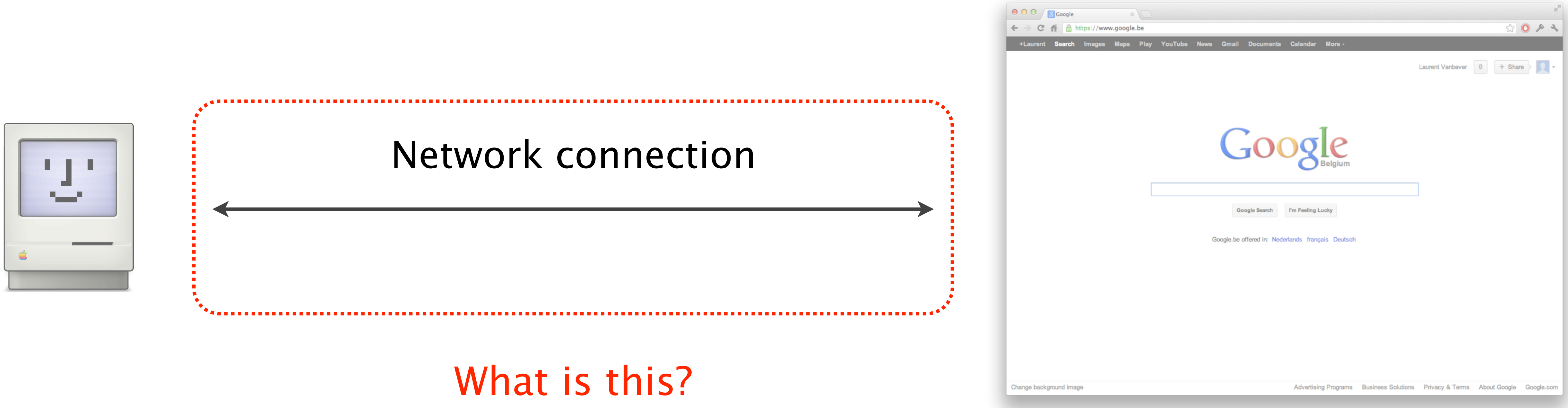
Google.be offered in: Nederlands français Deutsch

Network connection



SRC	Google
DST	Laurent

Payload

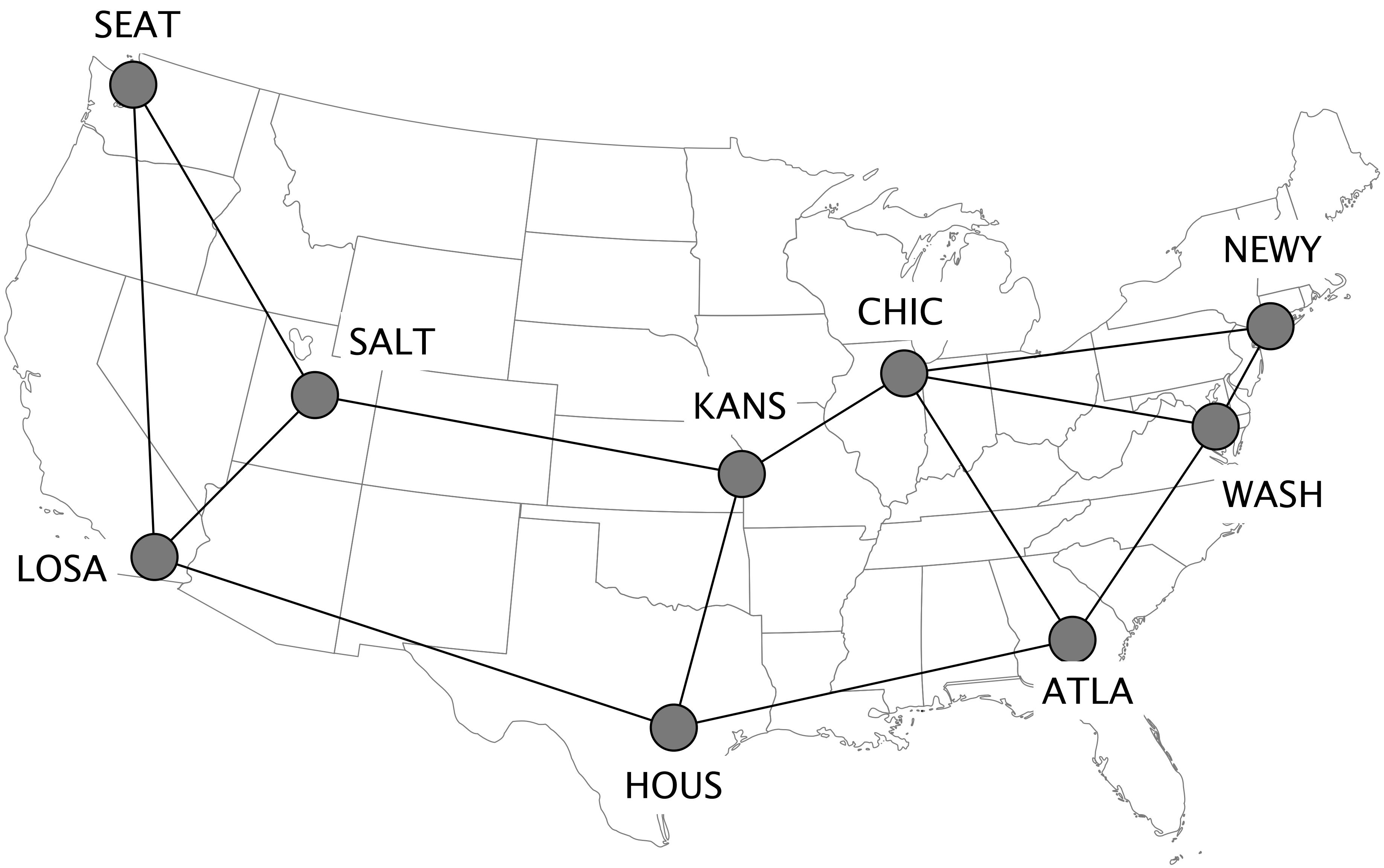


Network connection

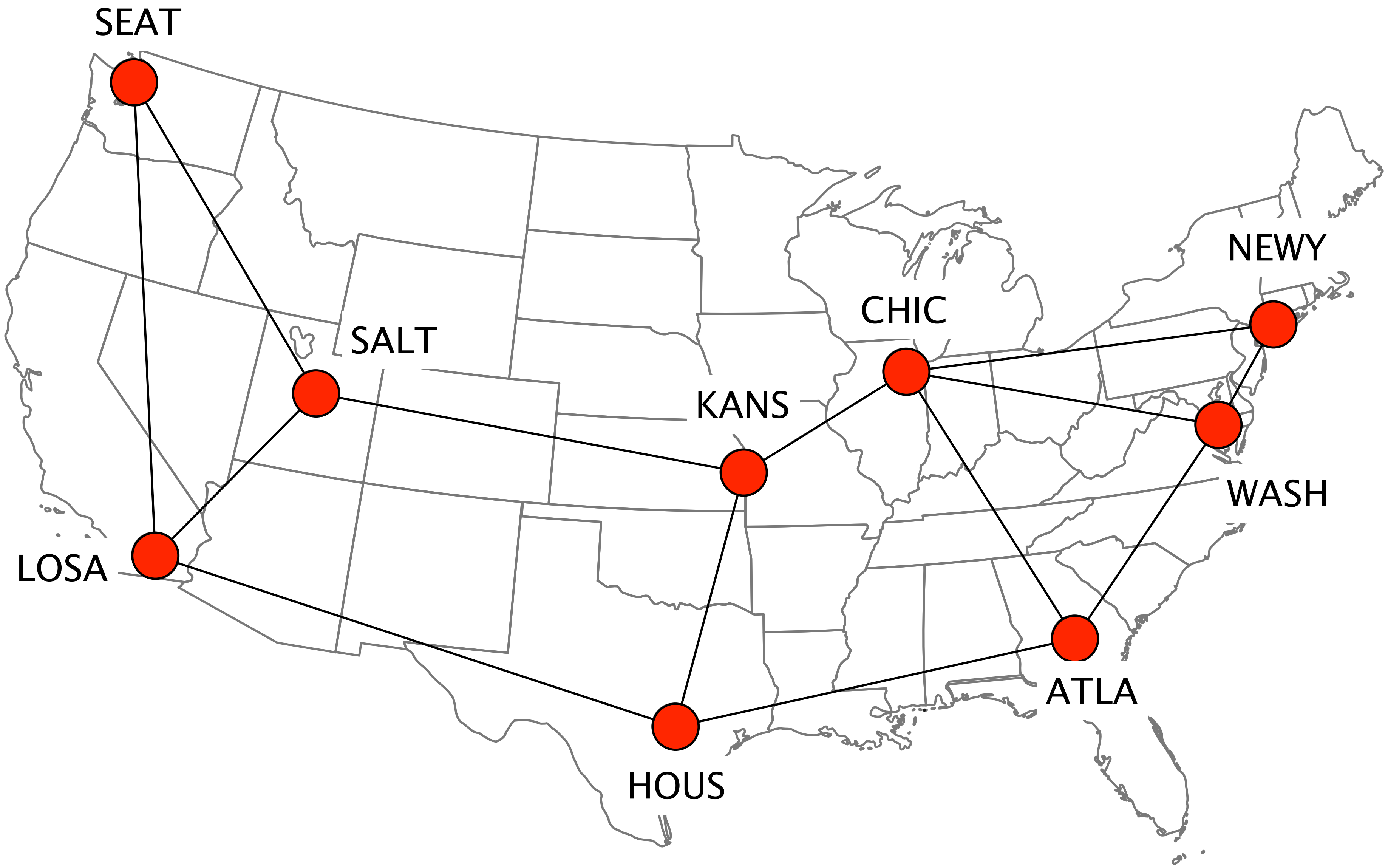
What is this?

A network is a distributed system

The US research network (Abilene, Internet2)



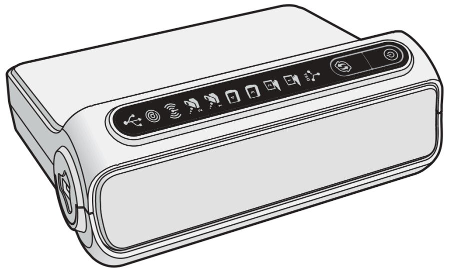
A network is a distributed system composed of routers



A network is a distributed system composed of routers

Internet core router

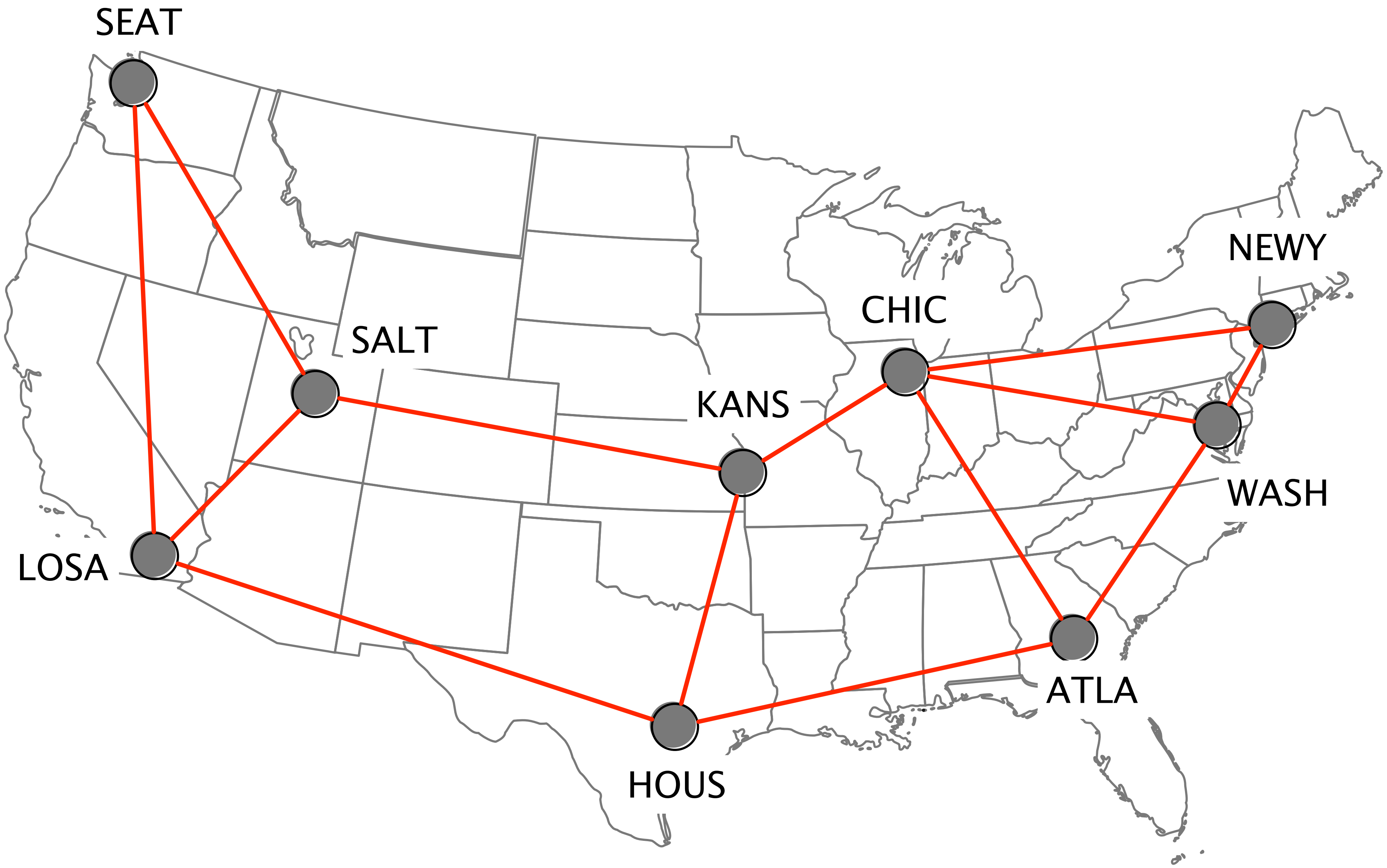
home routers



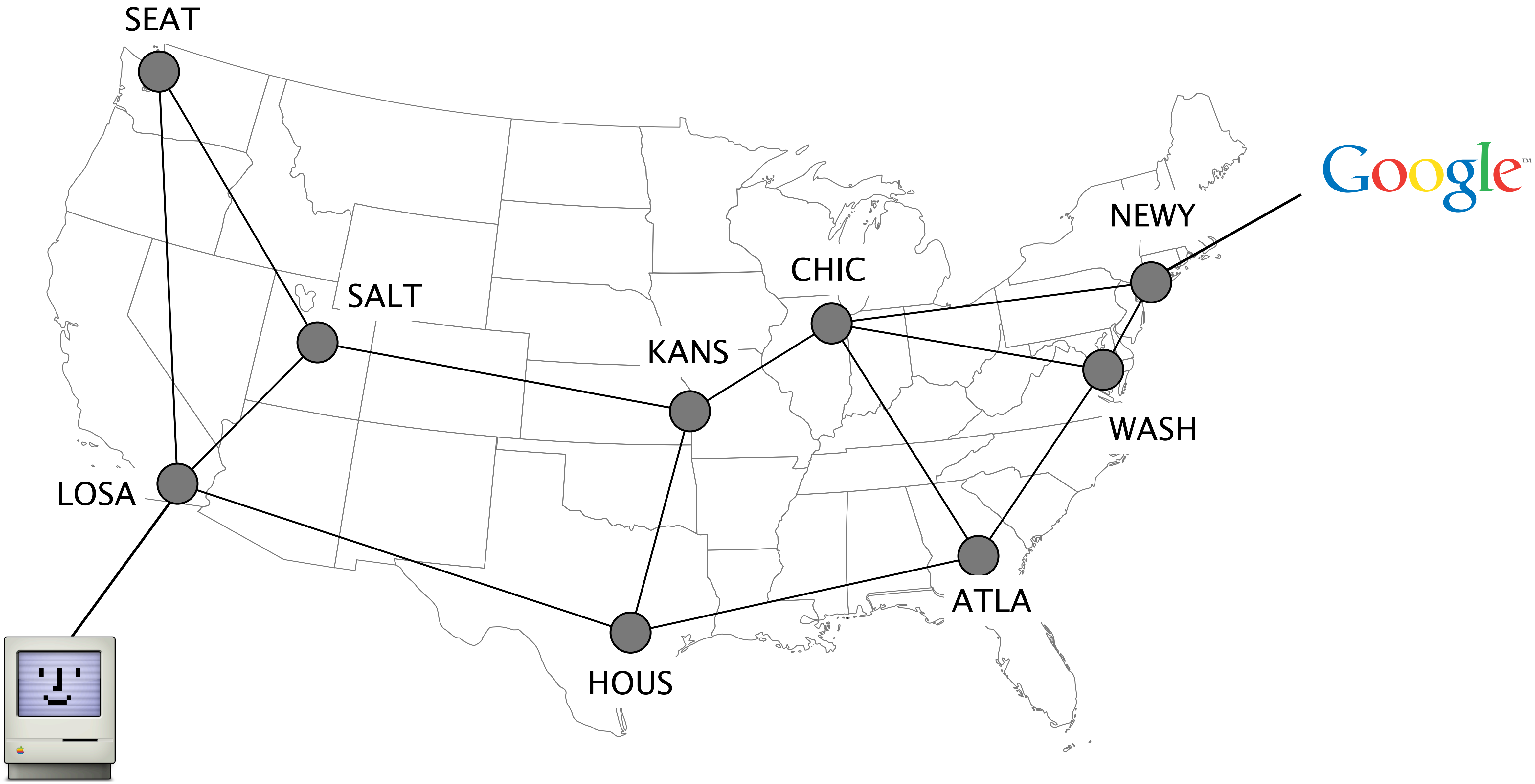
~ 15cm, 0,5kg, 100Mbps

>200cm, 700kg, 1.2 Tbps

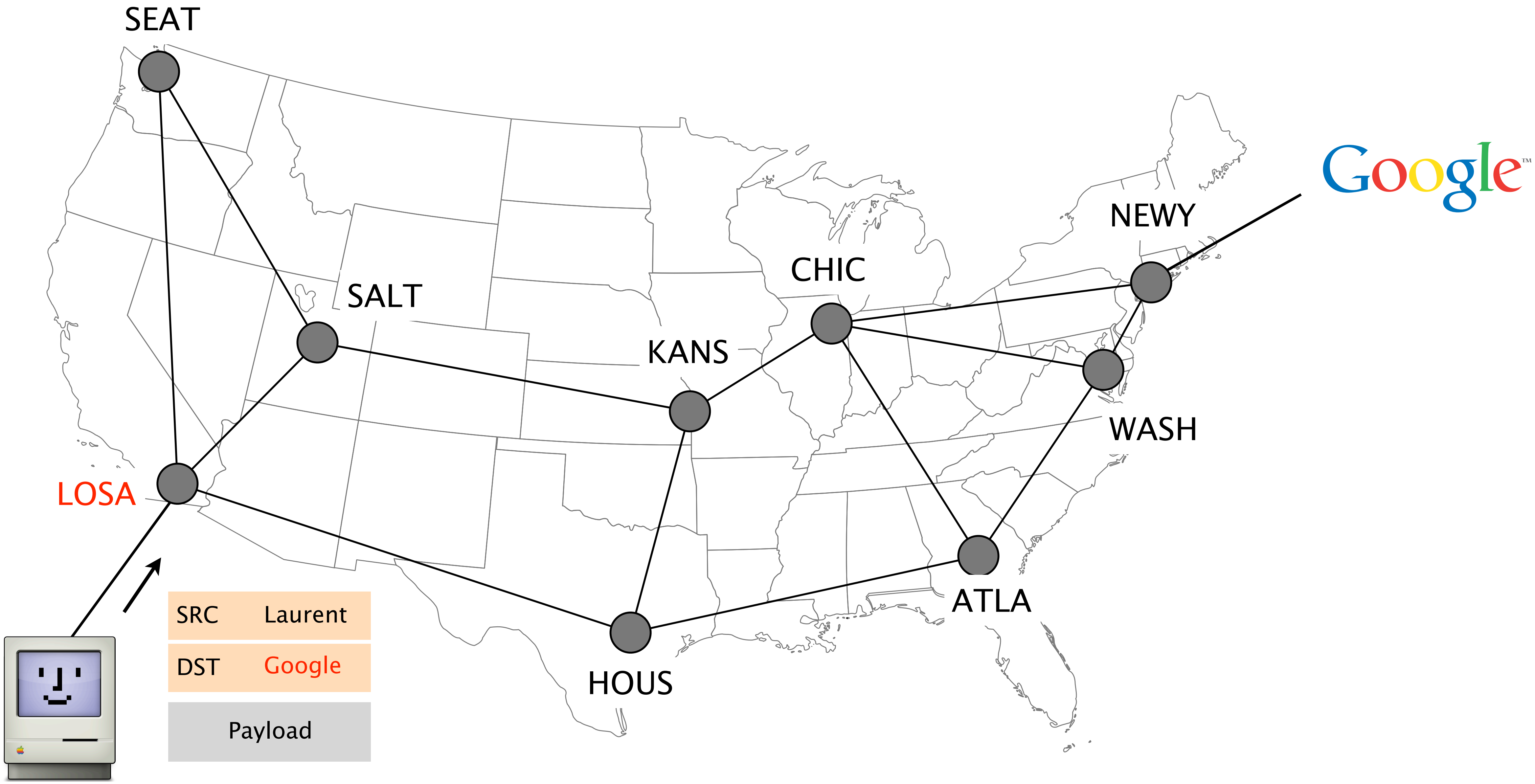
A network is a distributed system composed of routers and links



Routers forward IP packets towards their destination



Routers forward IP packets towards their destination



SEAT

SALT

KANS

CHIC

NEWY

WASH

ATLA

HOUS

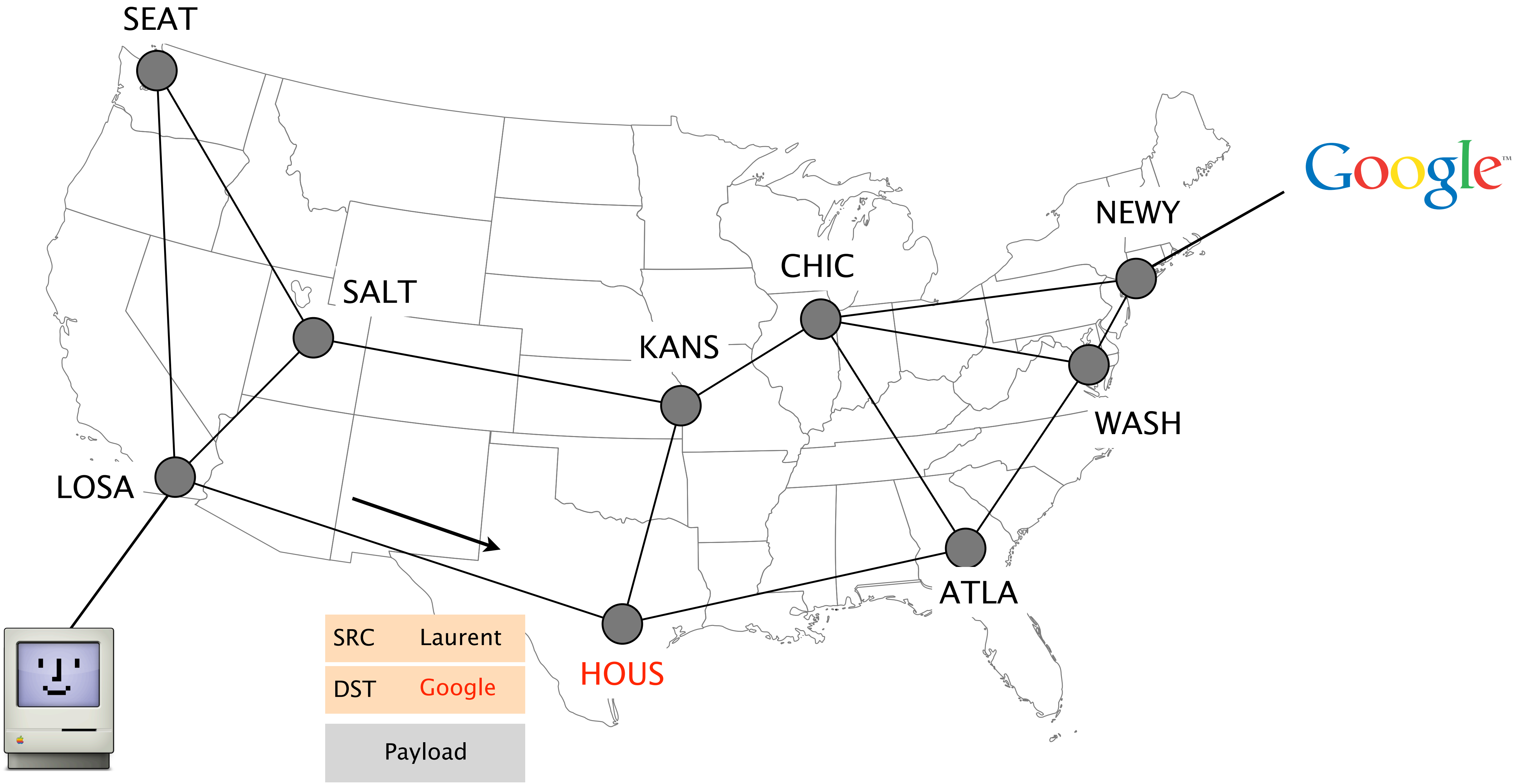
LOSA

Google

SRC	Laurent
DST	Google

Payload

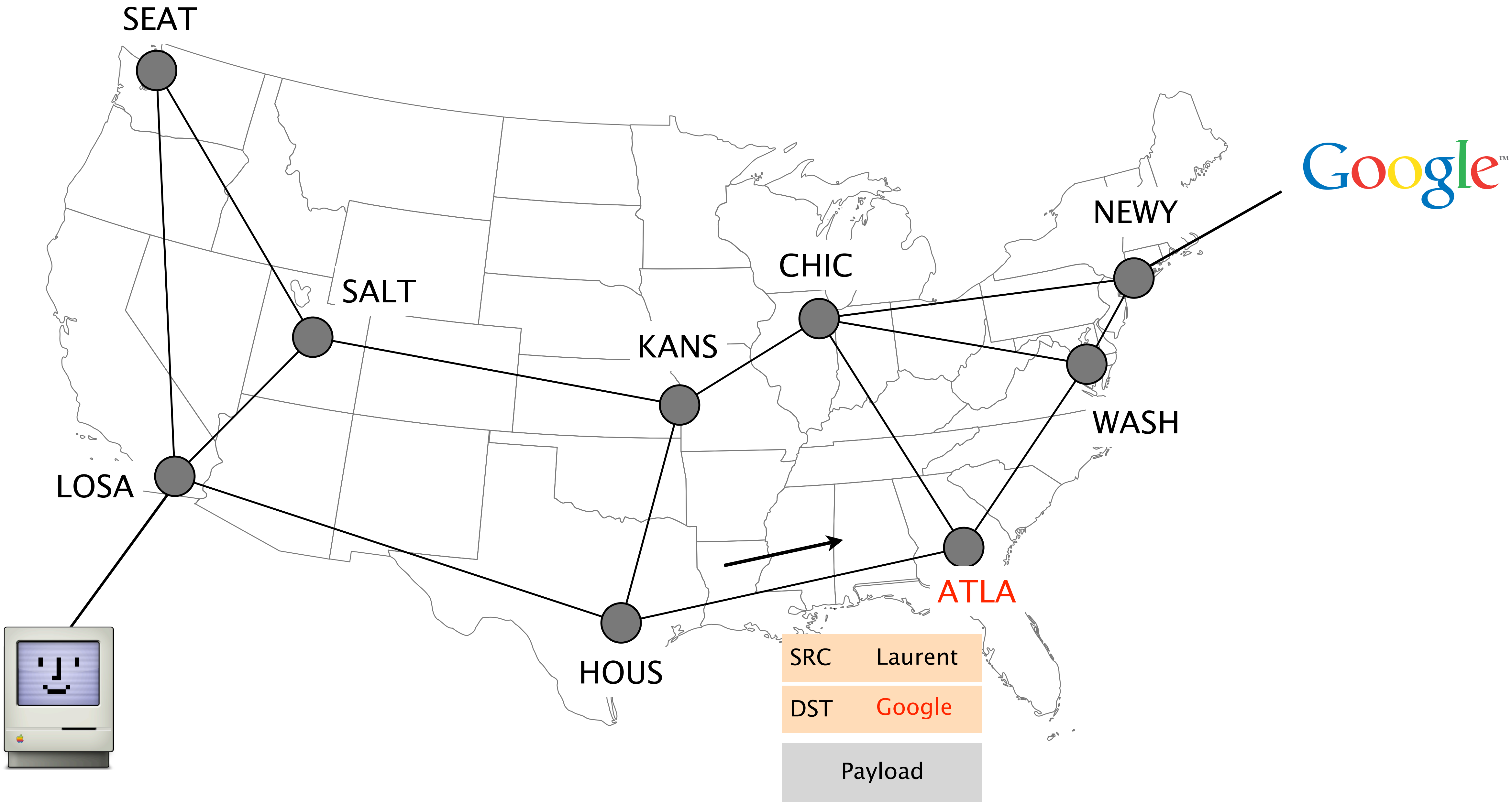
Routers forward IP packets towards their destination



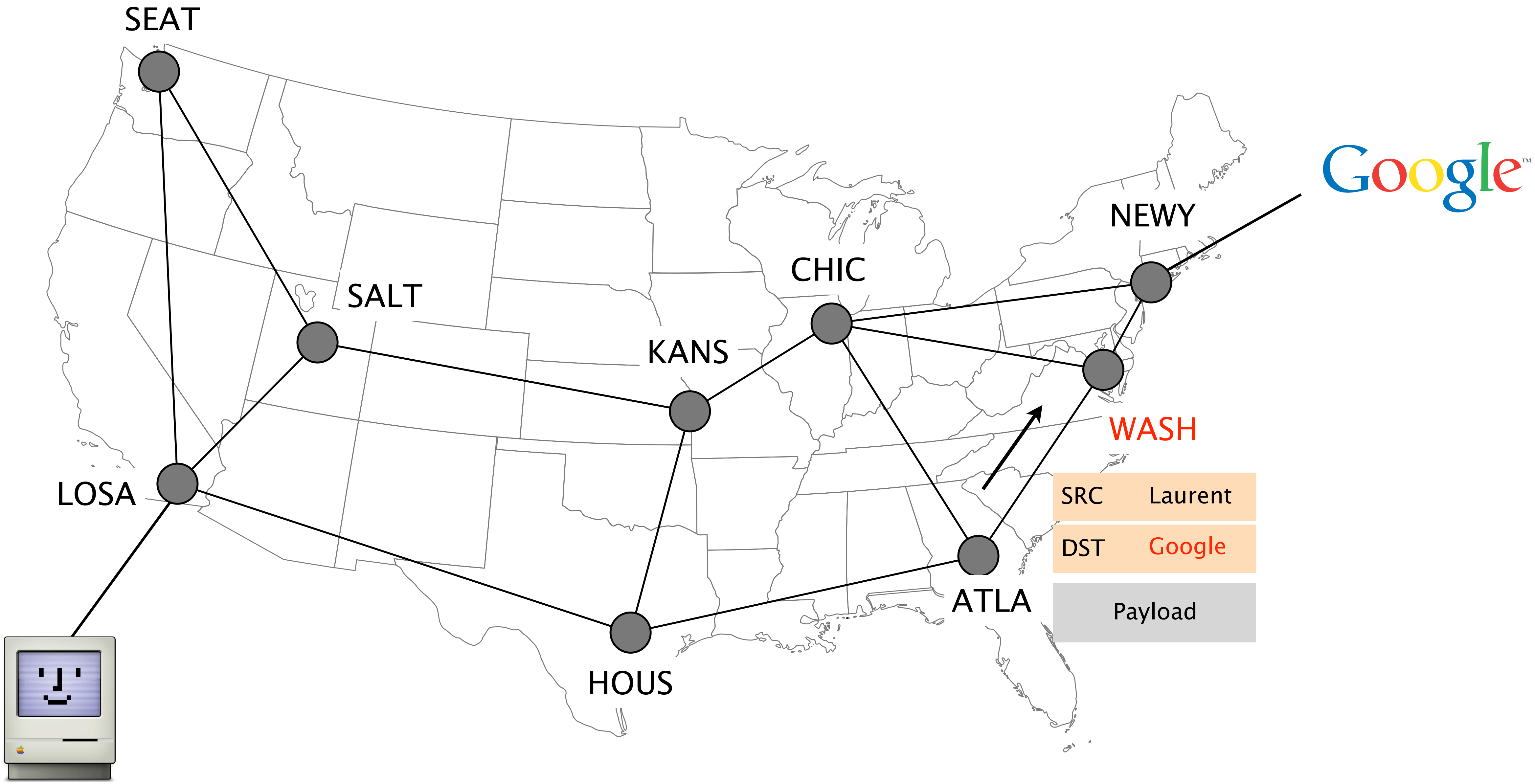
SRC	Laurent
DST	Google

Payload

Routers forward IP packets towards their destination



Routers forward IP packets towards their destination

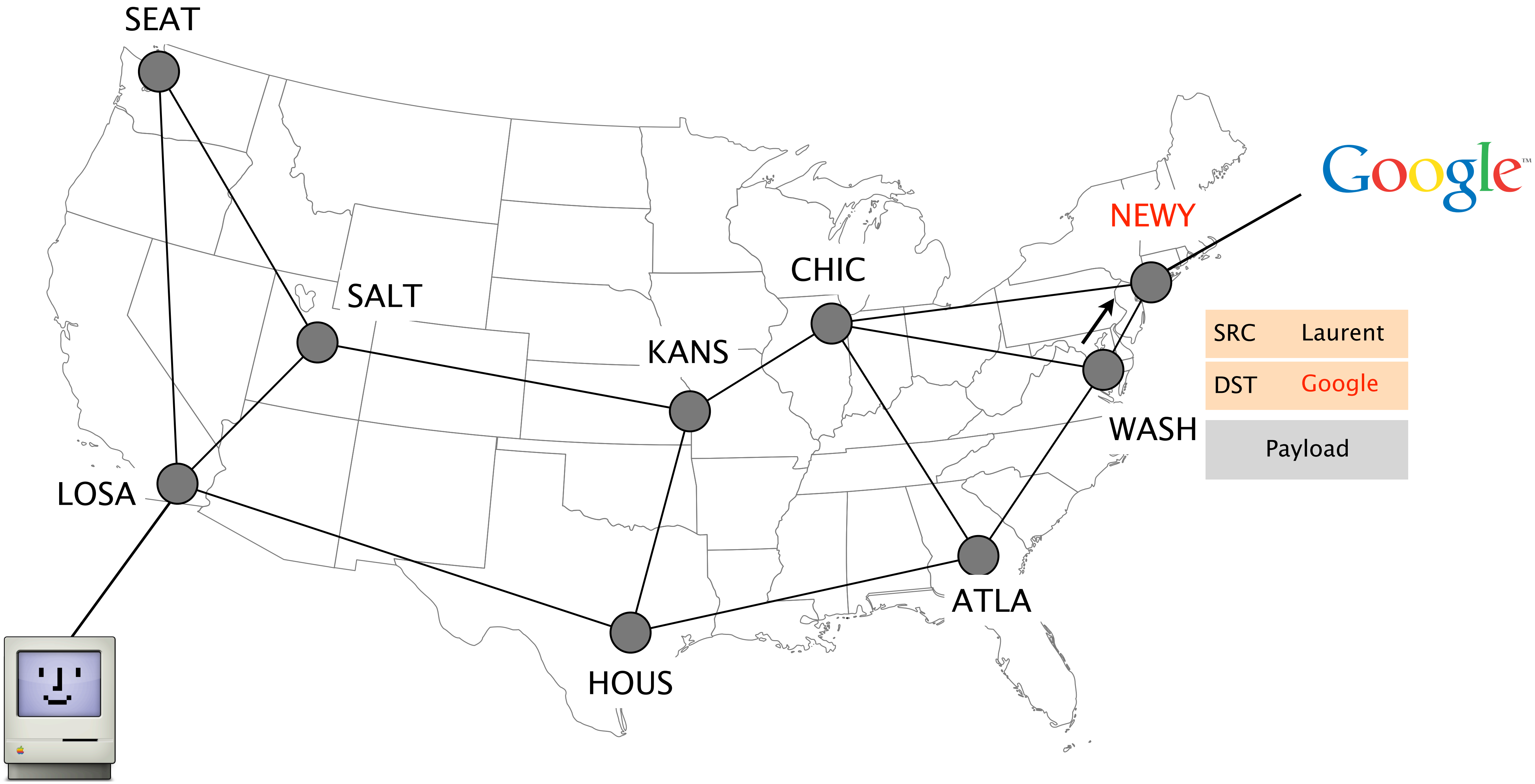


SRC Laurent

DST Google

Payload

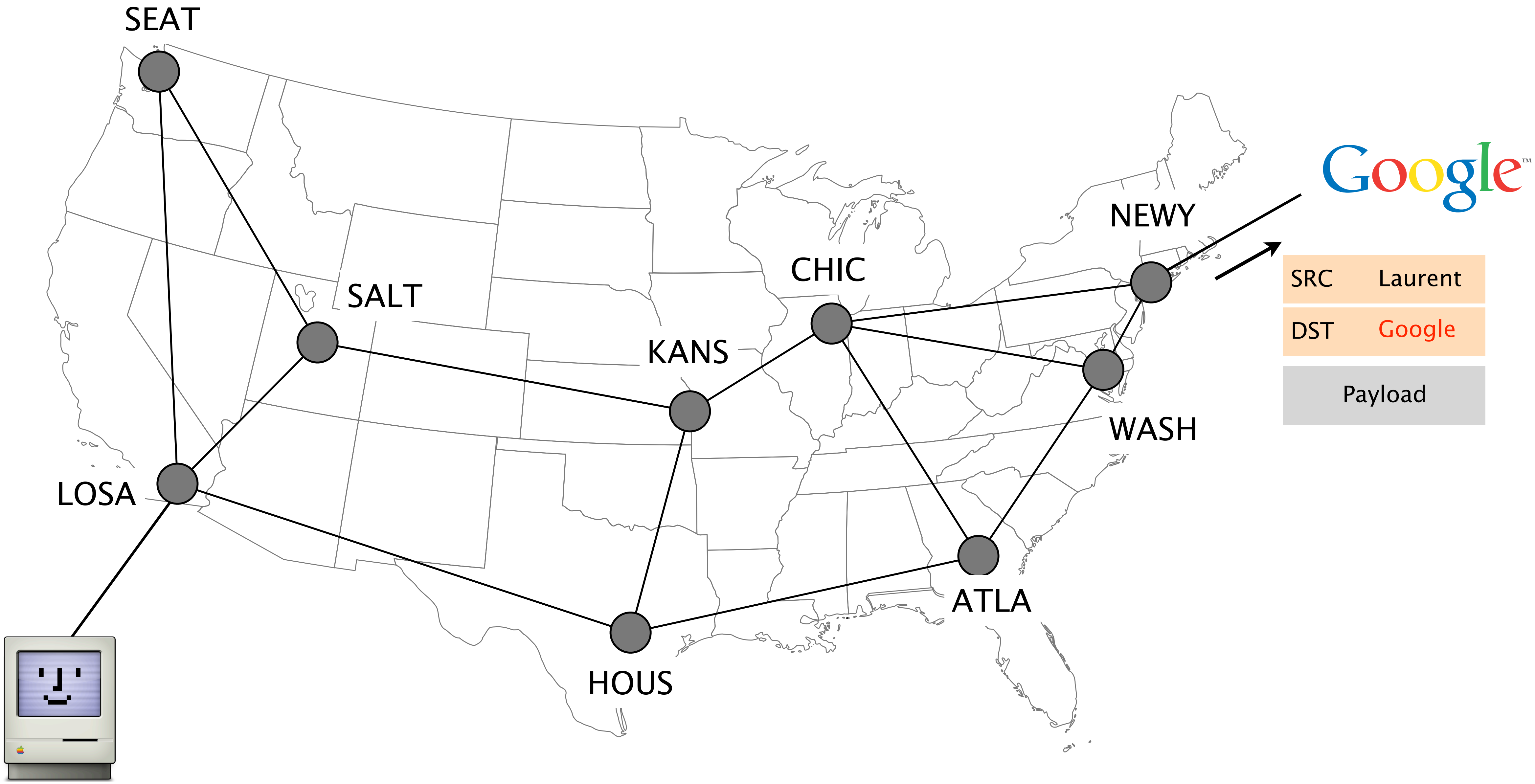
Routers forward IP packets towards their destination



SRC	Laurent
DST	Google

Payload

Routers forward IP packets towards their destination

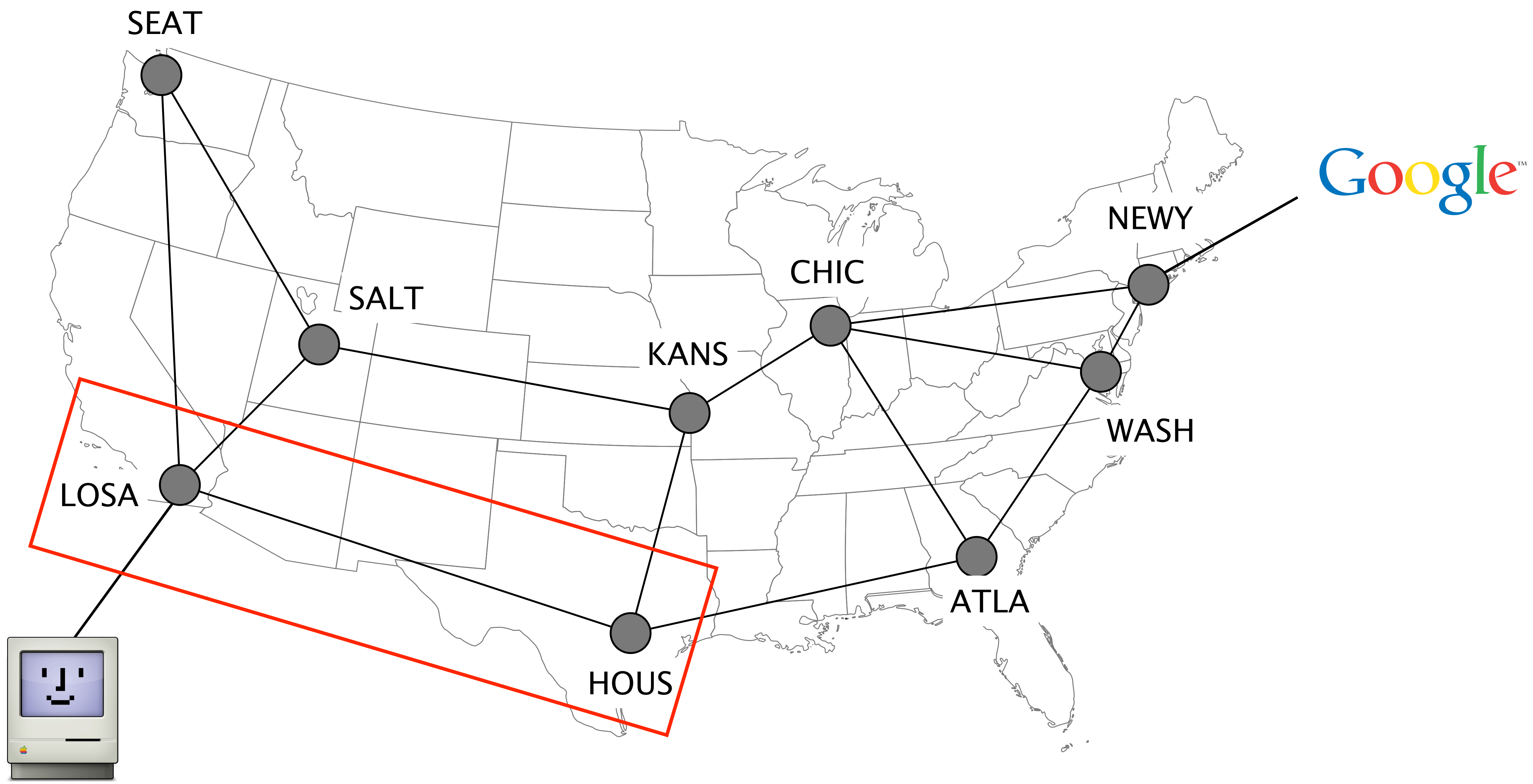


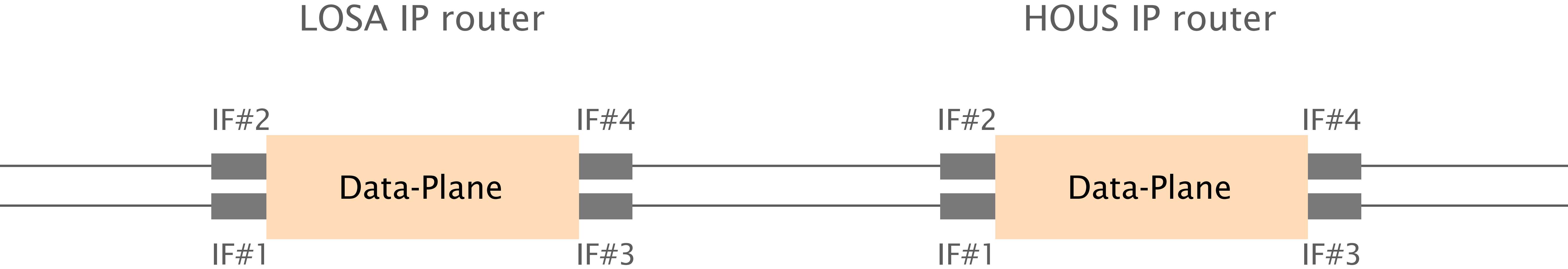
SRC Laurent

DST Google

Payload

Routers forward IP packets towards their destination





LOSA IP router

IF#2

IF#4

Data-Plane

IF#1

IF#3

HOUS IP router

IF#2

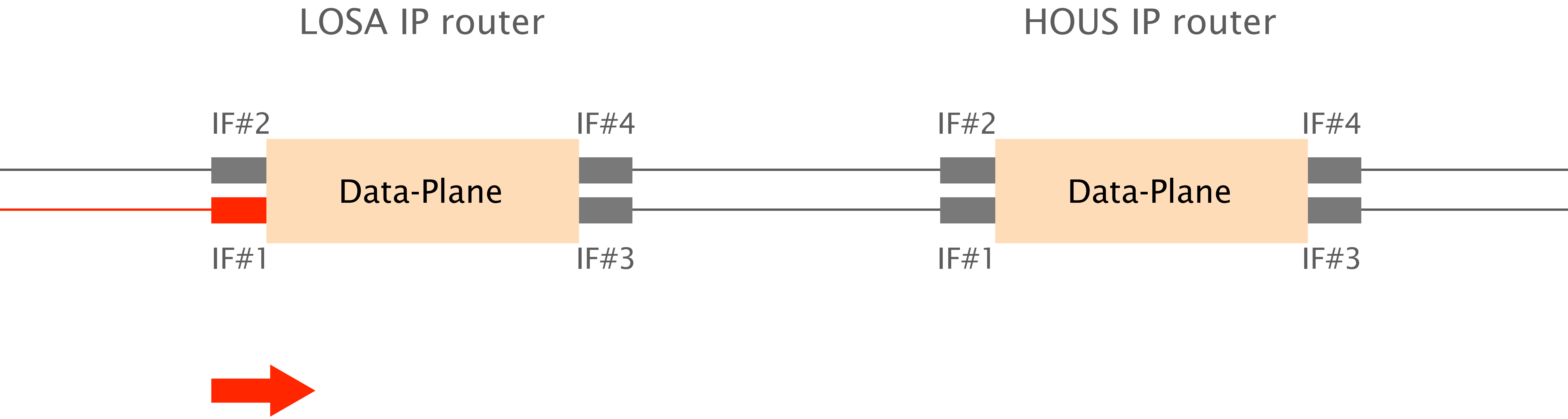
IF#4

Data-Plane

IF#1

IF#3

To forward an IP packet,
a router uses its routing table



LOSA IP router

HOUS IP router

IF#2

IF#4

Data-Plane

IF#1

IF#3

IF#2

IF#4

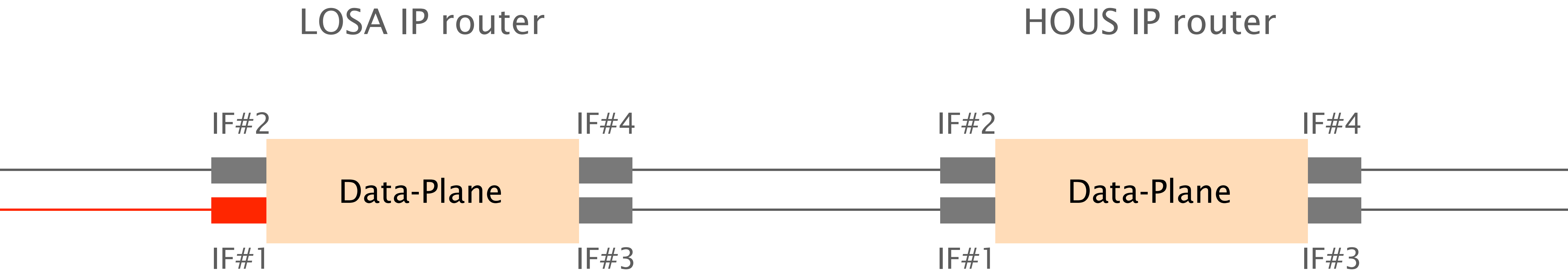
Data-Plane

IF#1

IF#3

SRC	Laurent
DST	Google
Payload	

To forward an IP packet,
a router uses its routing table

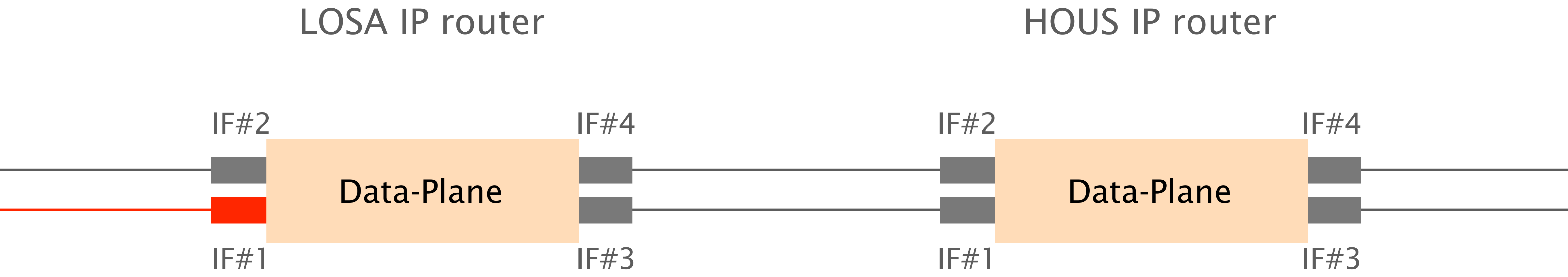


SRC	Laurent
DST	Google
Payload	

IP routing table

destination	output interface
Laurent	IF#1
Google	IF#4
…	…

To forward an IP packet, a router uses its routing table

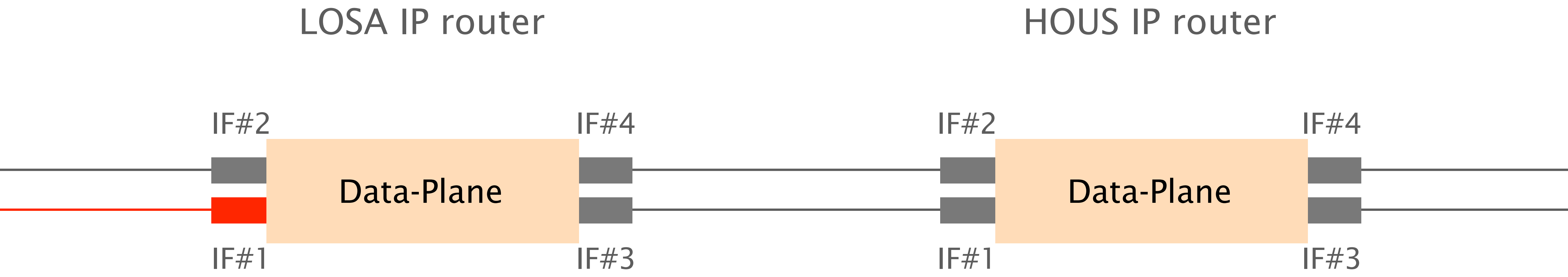


SRC	Laurent
DST	**Google**
Payload	

IP routing table

destination	output interface
Laurent	IF#1
Google	IF#4
…	…

To forward an IP packet, a router uses its routing table



LOSA IP router

IF#2 IF#4

Data-Plane

IF#1 IF#3

HOUS IP router

IF#2 IF#4

Data-Plane

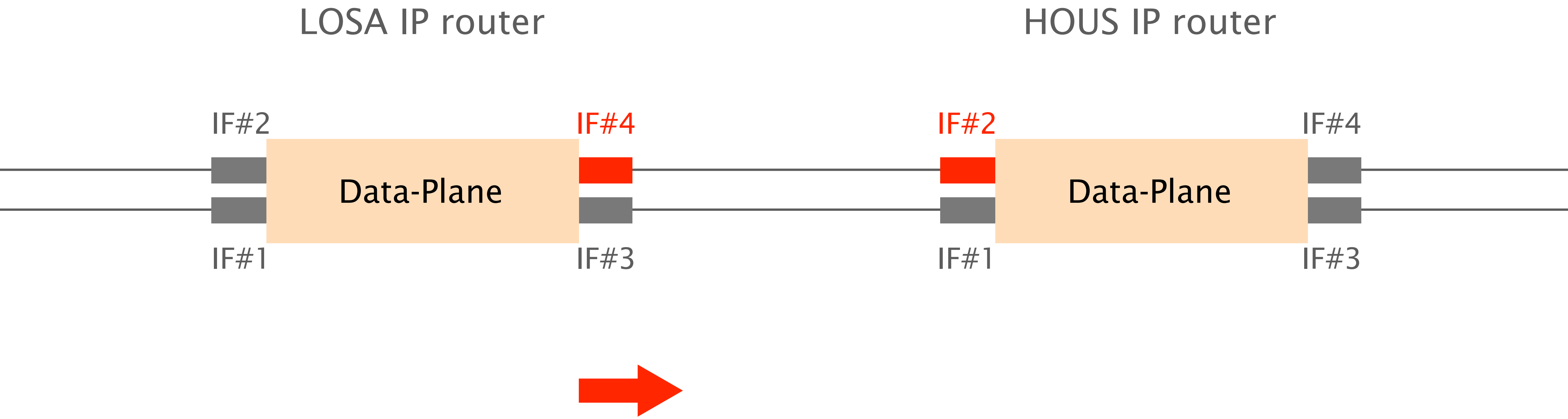
IF#1 IF#3

SRC	Laurent
DST	**Google**
Payload	

IP routing table

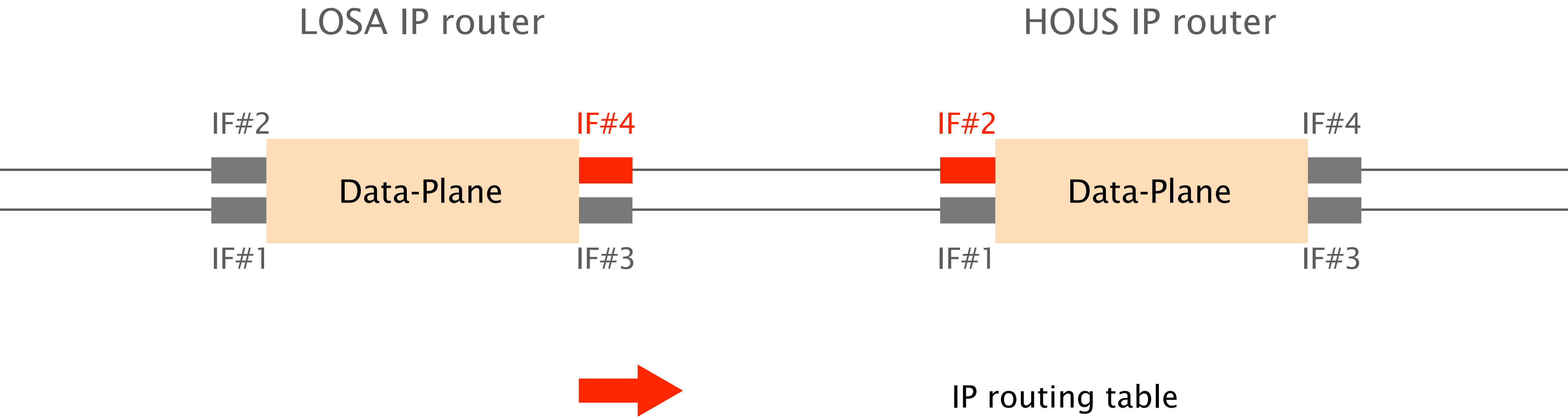
destination	output interface
Laurent	IF#1
Google	**IF#4**
...	...

To forward an IP packet,
a router uses its routing table



SRC	Laurent
DST	Google
Payload	

To forward an IP packet, a router uses its routing table

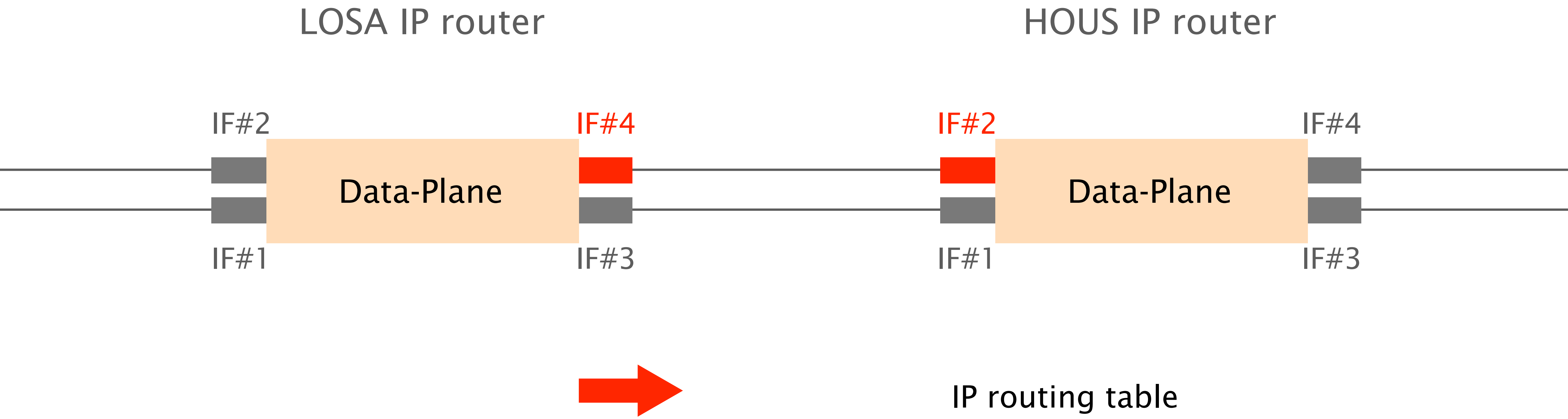


SRC	Laurent
DST	Google
Payload	

IP routing table

destination	output interface
Laurent	IF#1
Google	IF#3
…	…

To forward an IP packet, a router uses its routing table

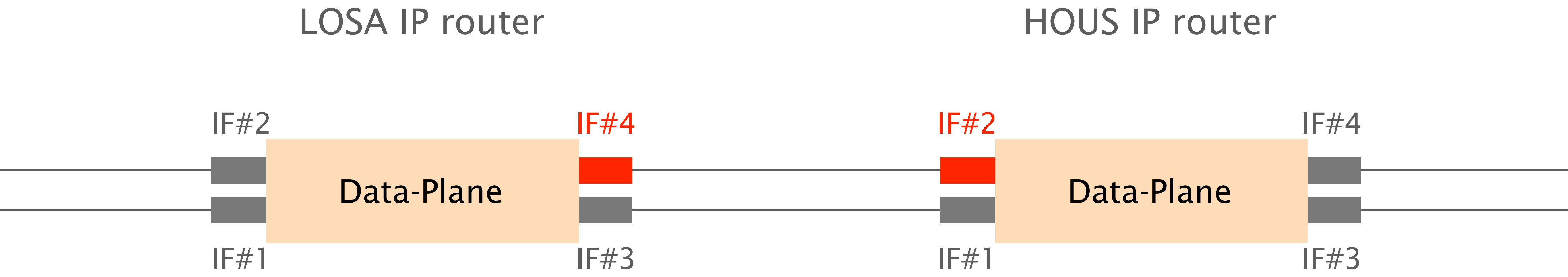


SRC	Laurent
DST	**Google**
Payload	

IP routing table

destination	output interface
Laurent	IF#1
Google	IF#3
…	…

To forward an IP packet,
a router uses its routing table

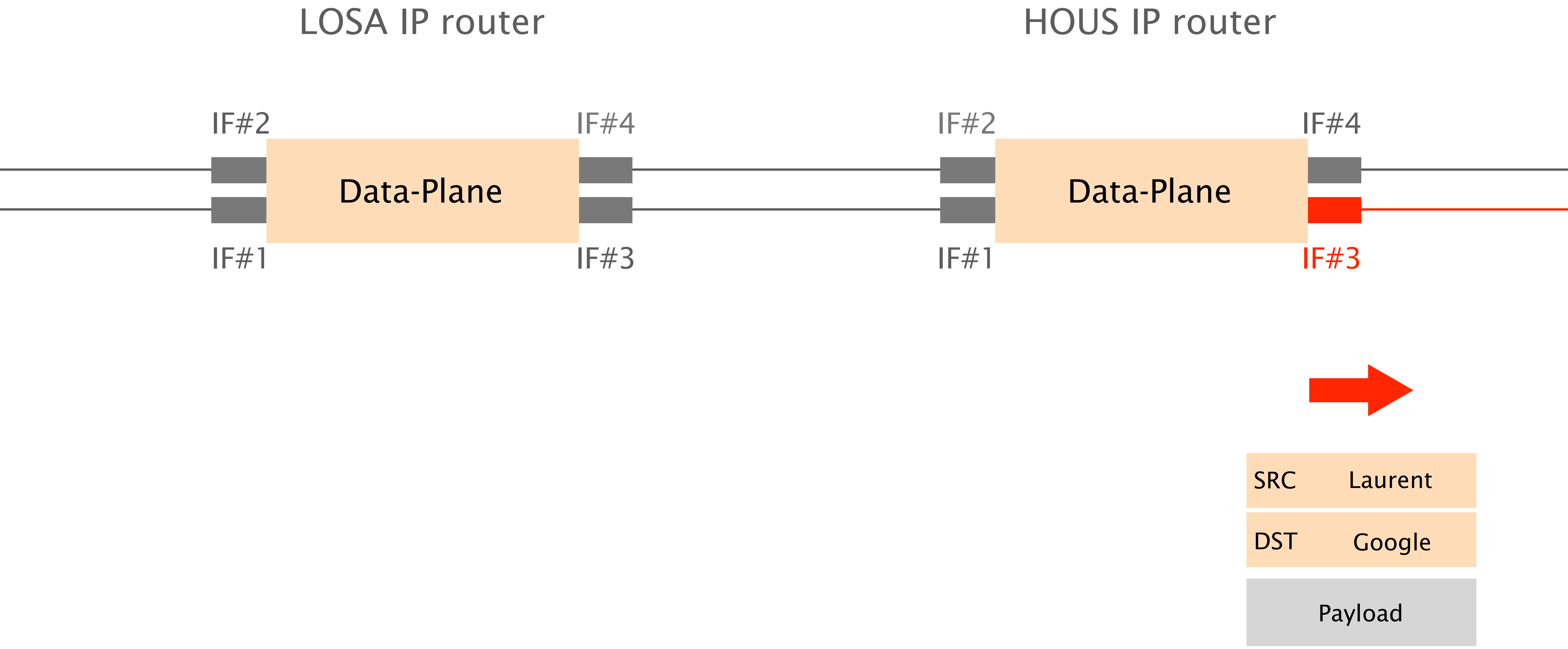


SRC	Laurent
DST	**Google**
Payload	

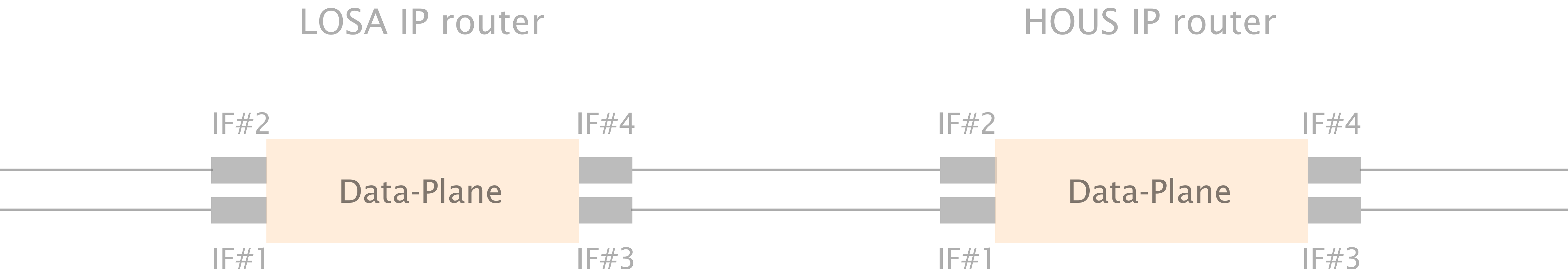
IP routing table

destination	output interface
Laurent	IF#1
Google	**IF#3**
...	...

To forward an IP packet, a router uses its routing table



How are computed the routing tables?

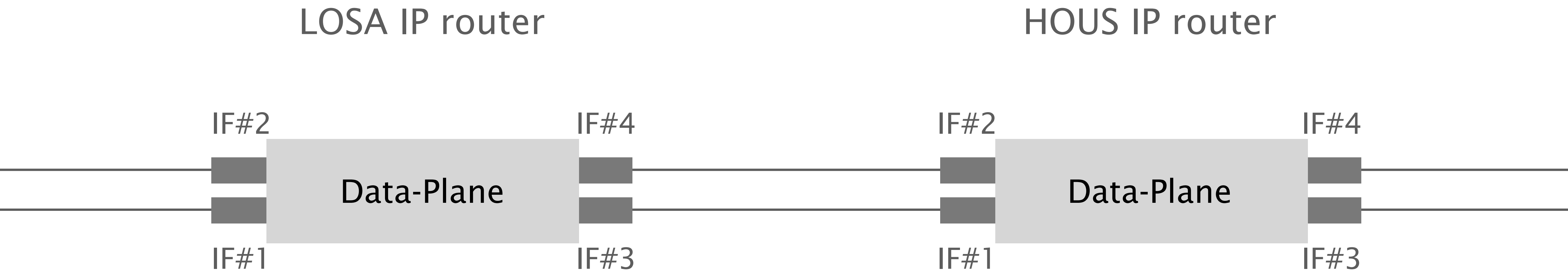


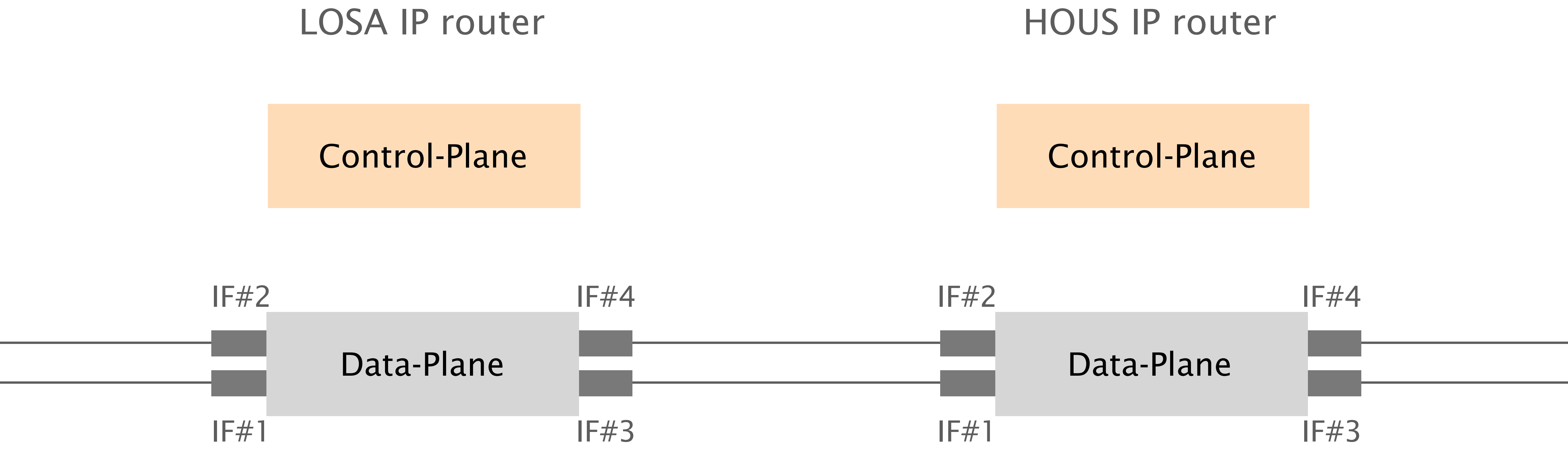
IP routing table

destination	output interface
Laurent	IF#1
Google	IF#4
…	…

IP routing table

destination	output interface
Laurent	IF#1
Google	IF#3
…	…


LOSA IP router
HOUS IP router
IF#2
IF#4
Data-Plane
IF#1
IF#3
IF#2
IF#4
Data-Plane
IF#1
IF#3



LOSA IP router

Control-Plane

IF#2

IF#4

Data-Plane

IF#1

IF#3

HOUS IP router

Control-Plane

IF#2

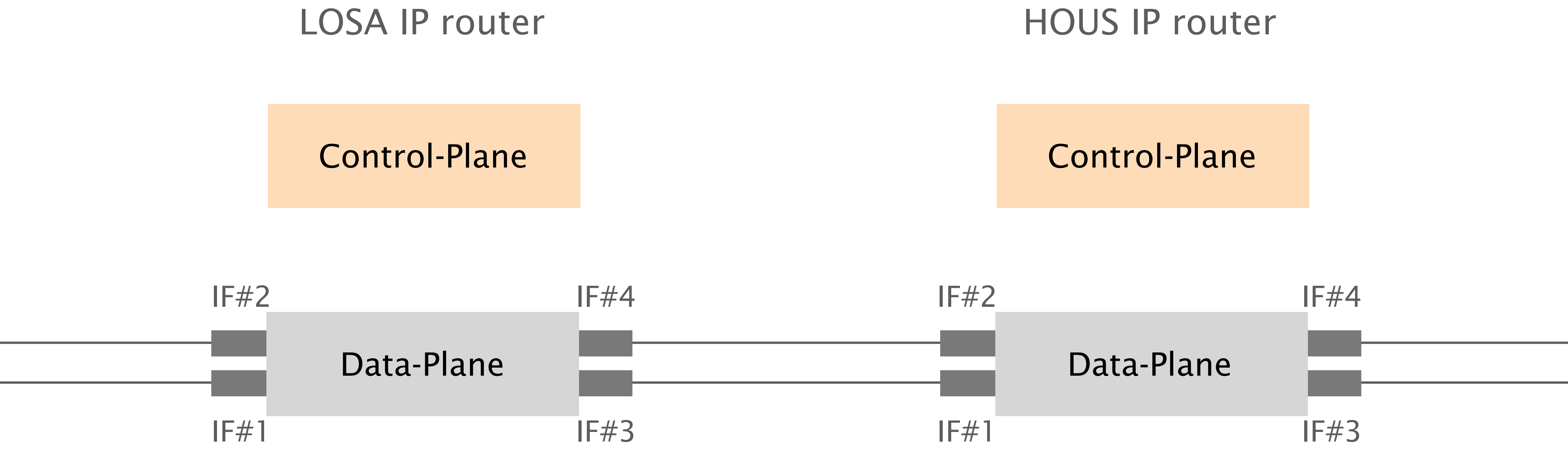
IF#4

Data-Plane

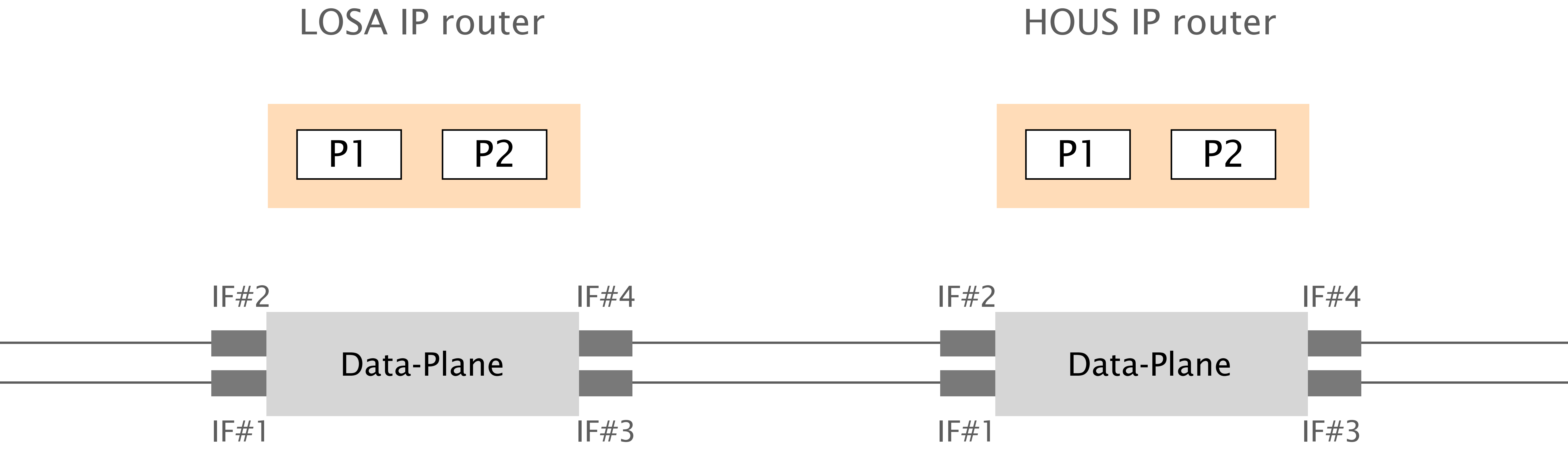
IF#1

IF#3

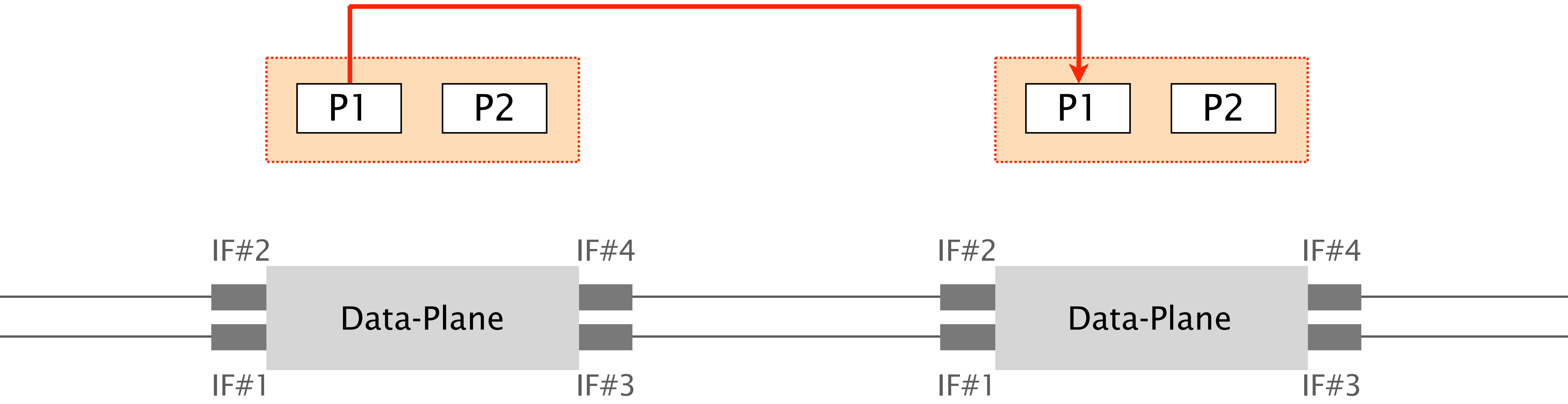
The control-plane advertises, learns and computes the routing table



To do so, the control-plane runs several routing protocols



These protocols exchange data about reachable destination

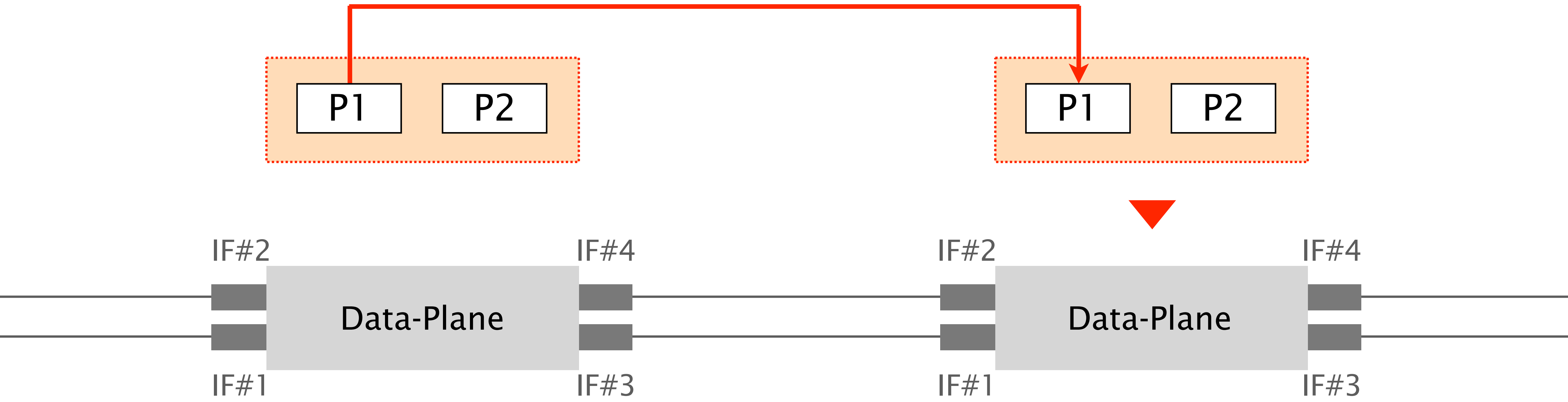

I can reach the destination “Laurent”

IP routing table

destination	output interface

These protocols exchange data about reachable destination

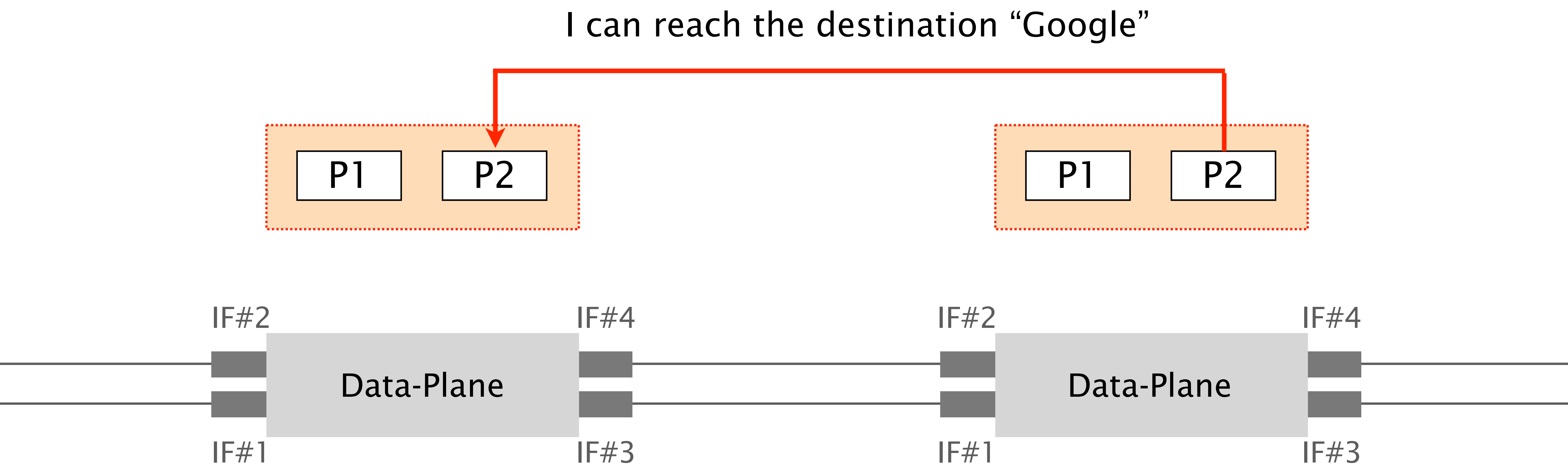
I can reach the destination “Laurent”



IP routing table

destination	output interface
Laurent	IF#1

These protocols exchange data about reachable destination

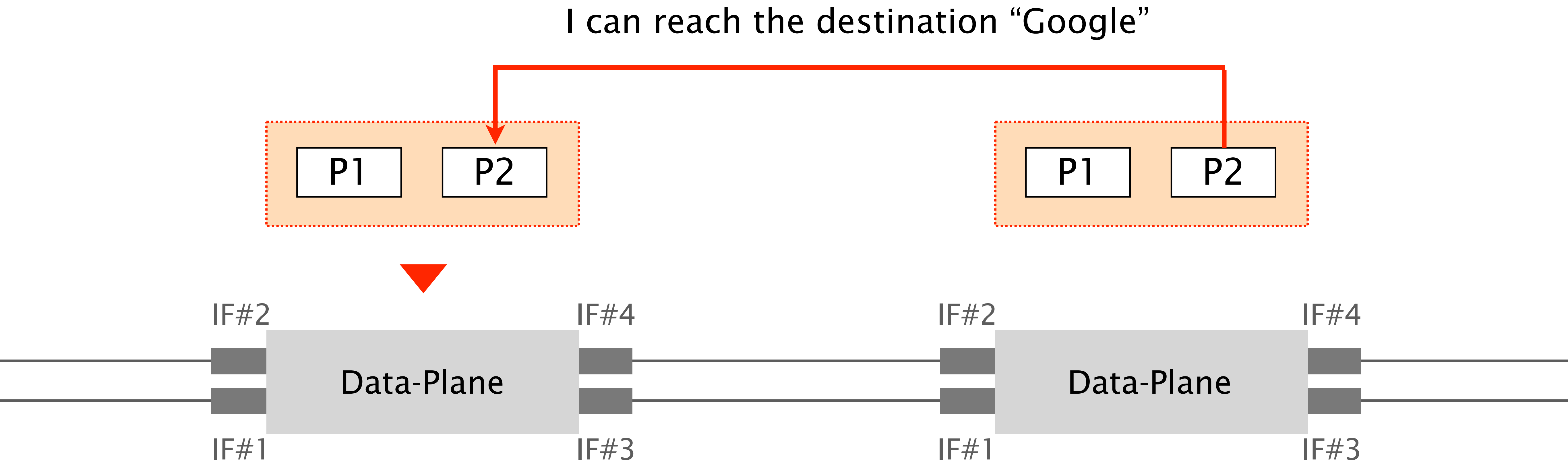


I can reach the destination “Google”

IP routing table

destination	output interface

These protocols exchange data about reachable destination

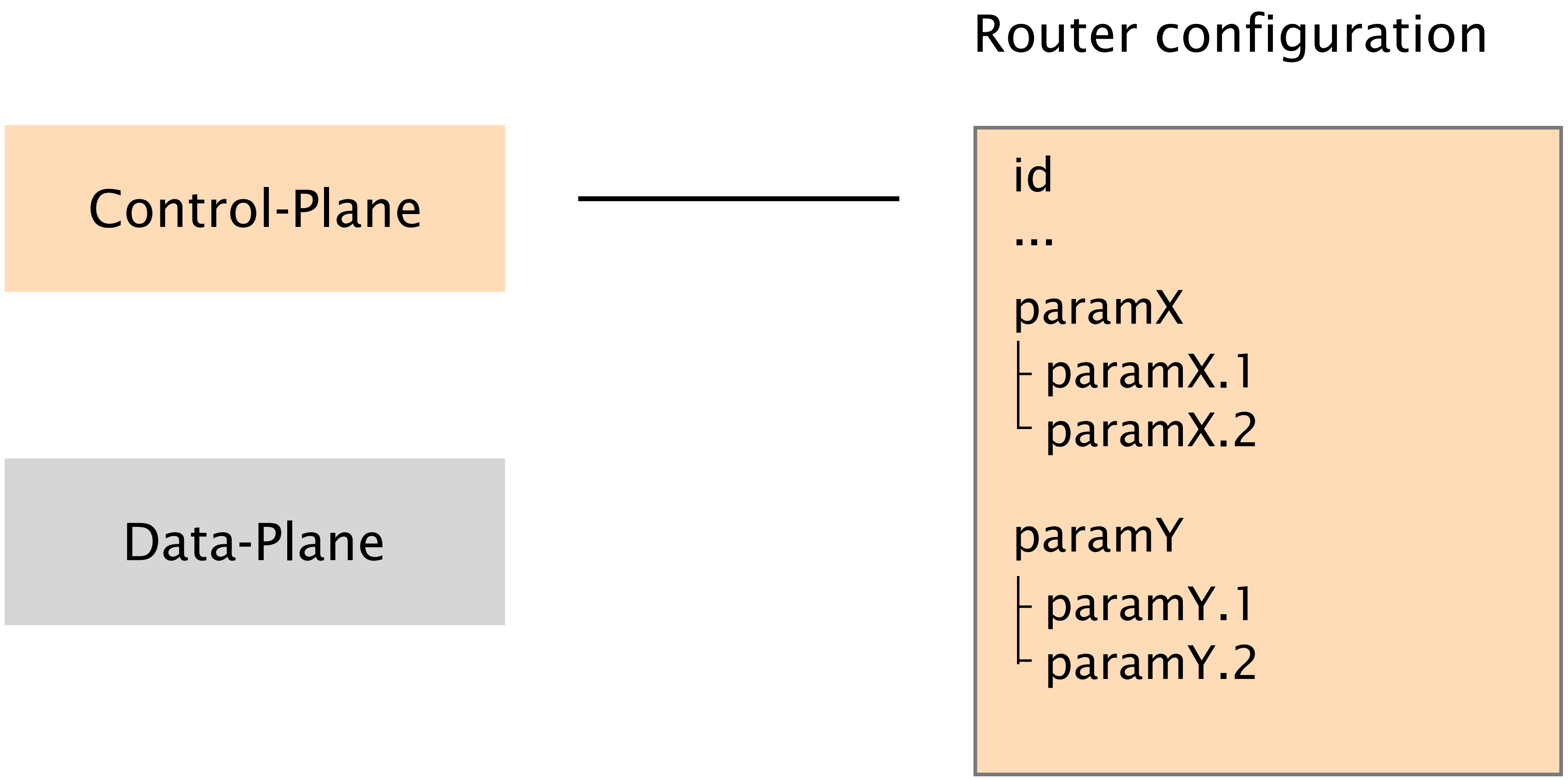


I can reach the destination “Google”

IP routing table

destination	output interface
Google	IF#4

The behavior of the routing protocols are defined by their configuration



Control-Plane

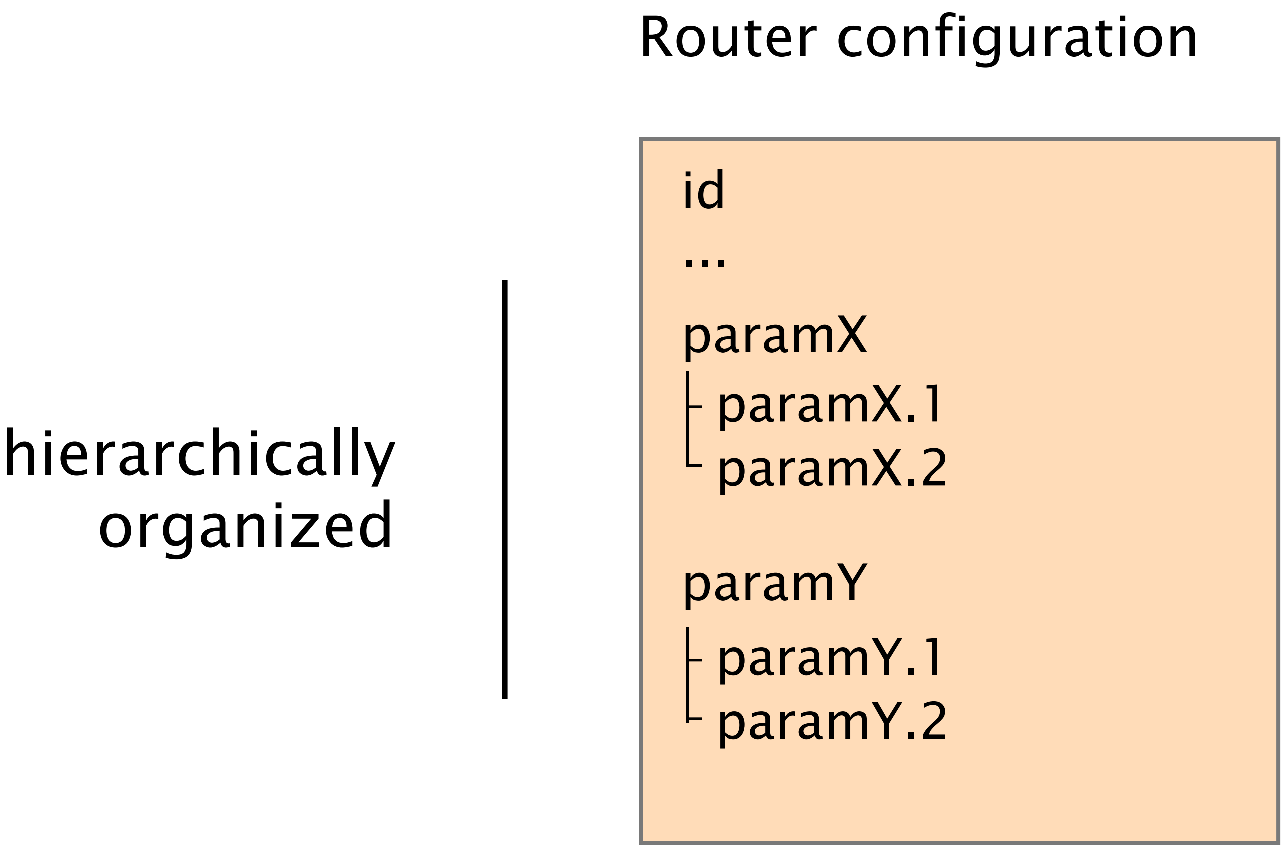
Data-Plane

Router configuration

```
id
...
paramX
├ paramX.1
└ paramX.2

paramY
├ paramY.1
└ paramY.2
```

A configuration is a set of parameters



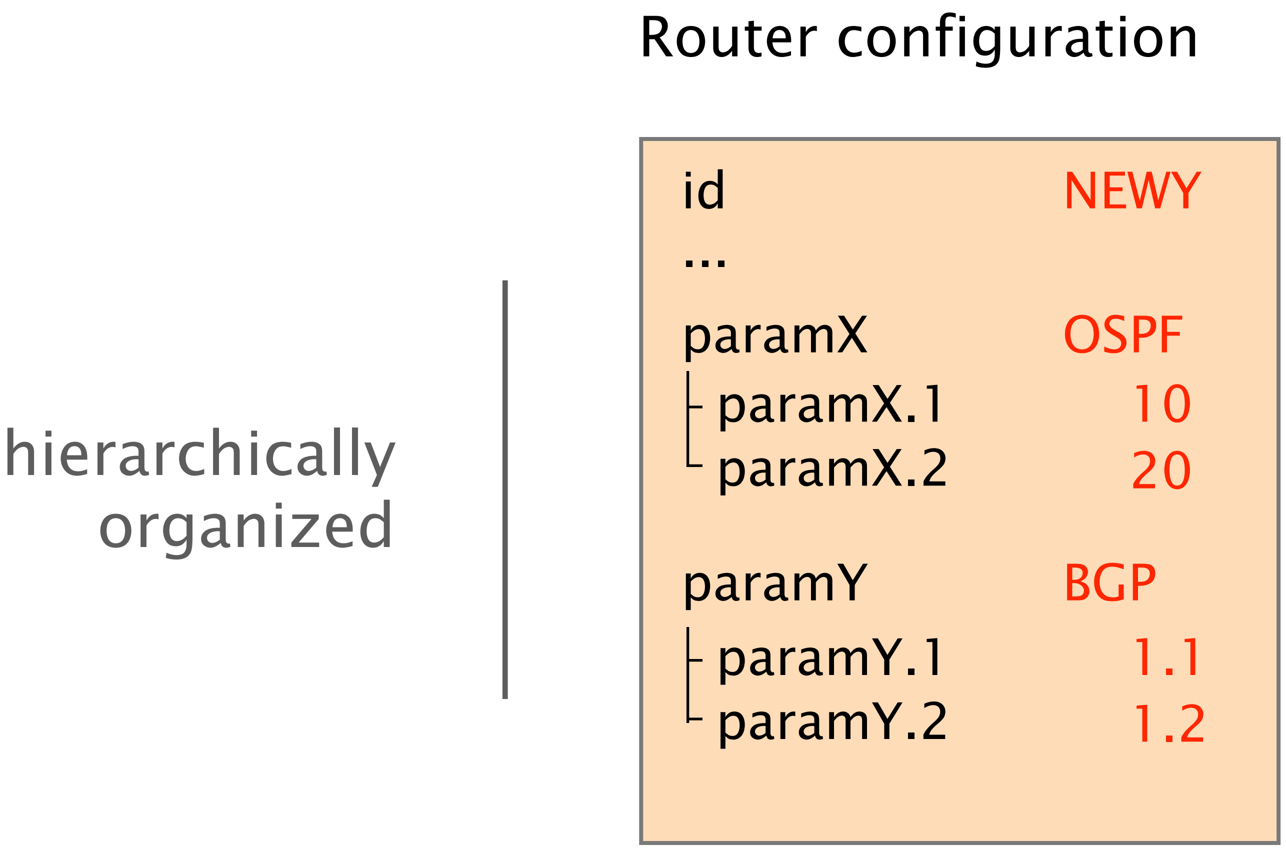
Router configuration

hierarchically organized

```
id
...
paramX
├ paramX.1
└ paramX.2

paramY
├ paramY.1
└ paramY.2
```

A configuration is a set of parameters **and values**

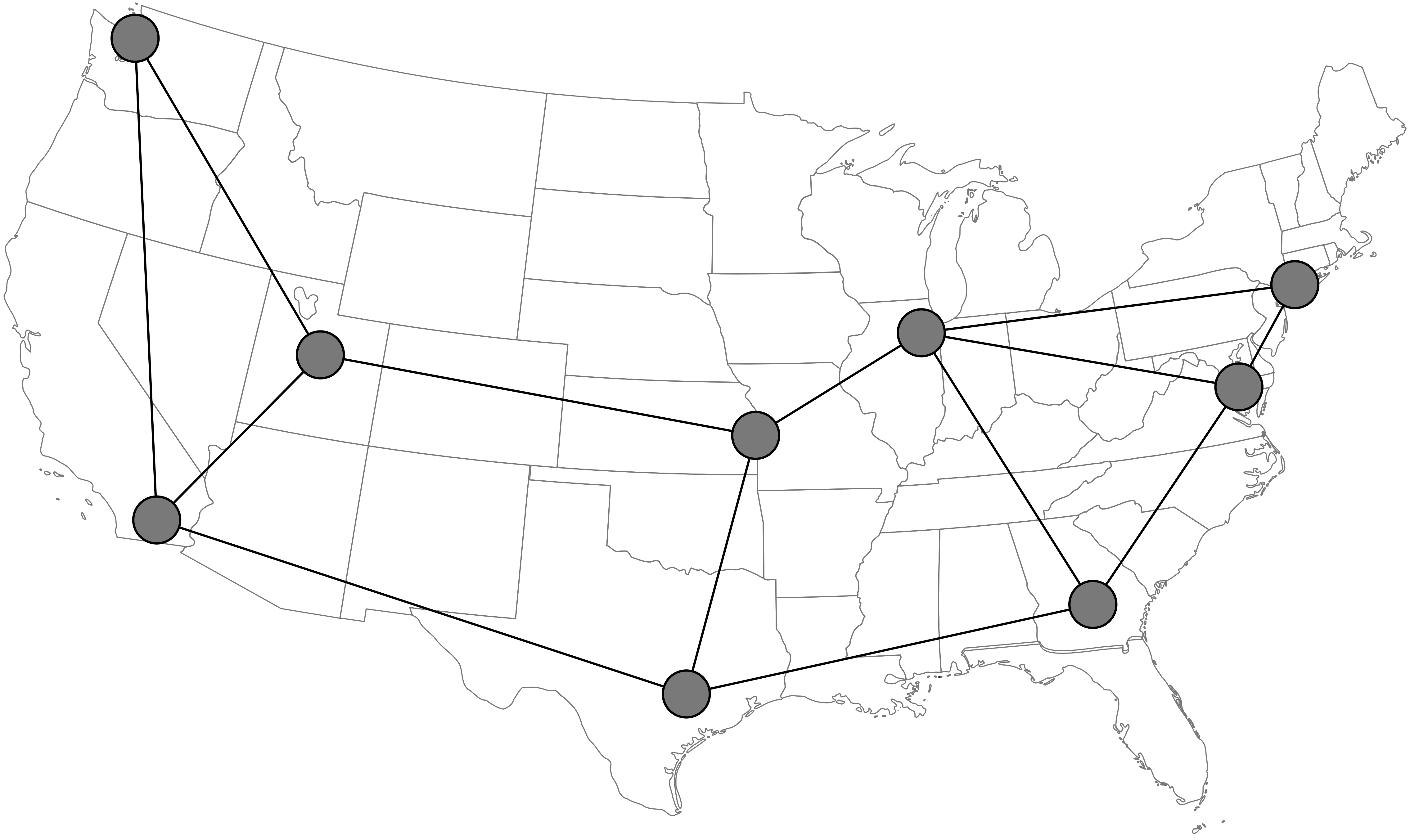


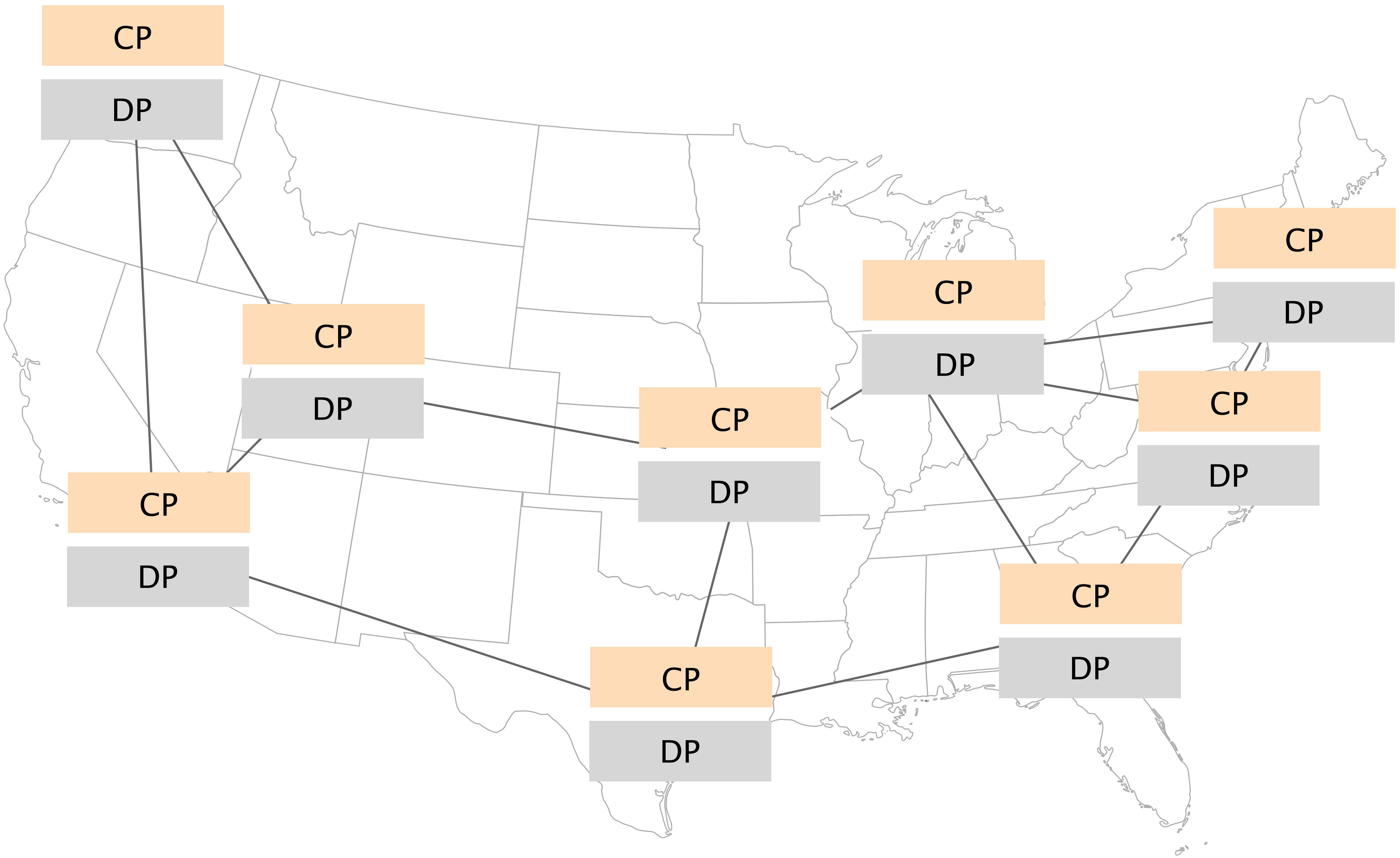
hierarchically organized

Router configuration

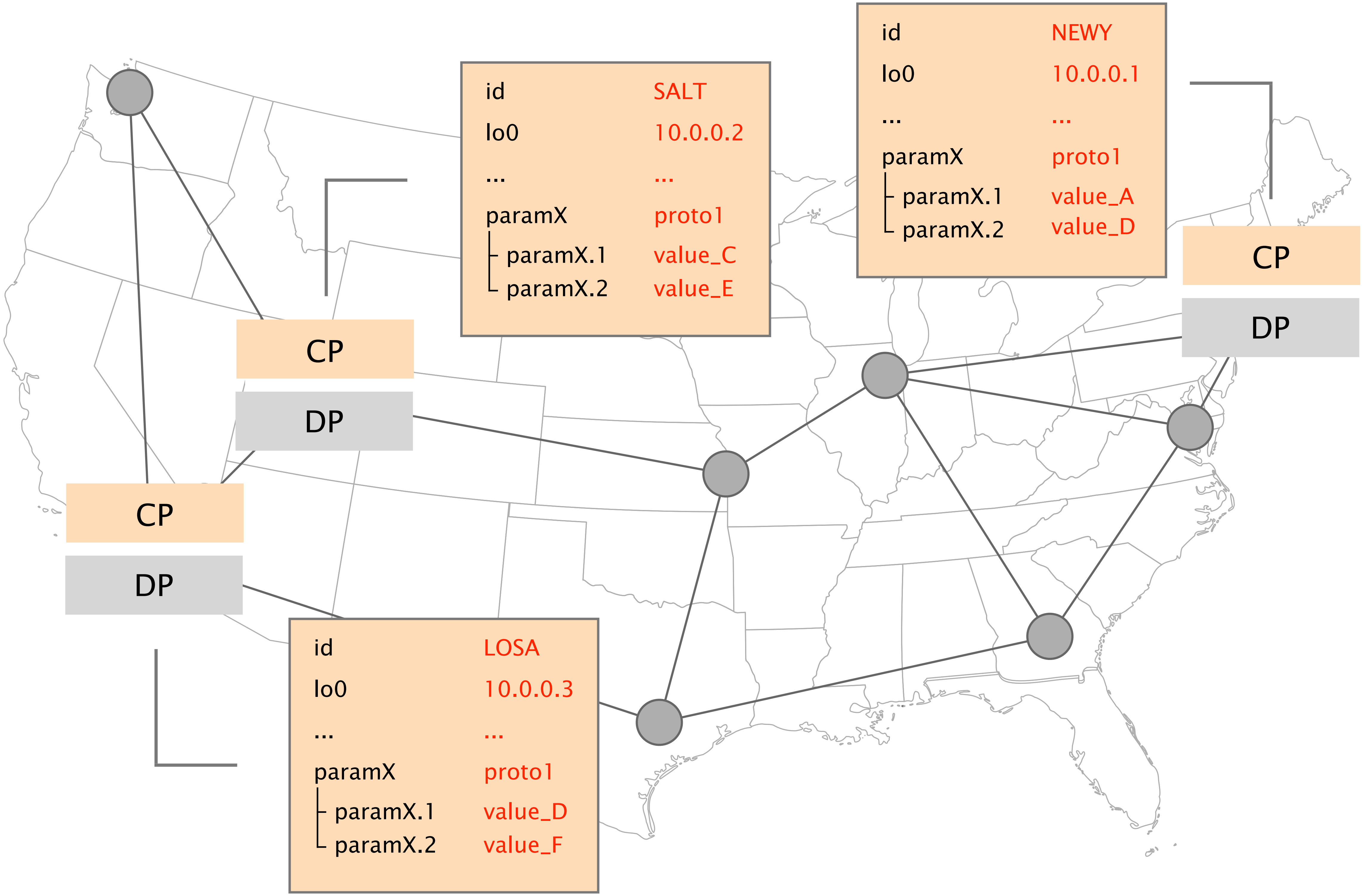
```
id              NEWY
...
paramX          OSPF
├ paramX.1        10
└ paramX.2        20

paramY          BGP
├ paramY.1        1.1
└ paramY.2        1.2
```




CP
DP
CP
DP
CP
DP
CP
DP
CP
DP
CP
DP
CP
DP
CP
DP
CP
DP

A network is a distributed system with a distributed configuration

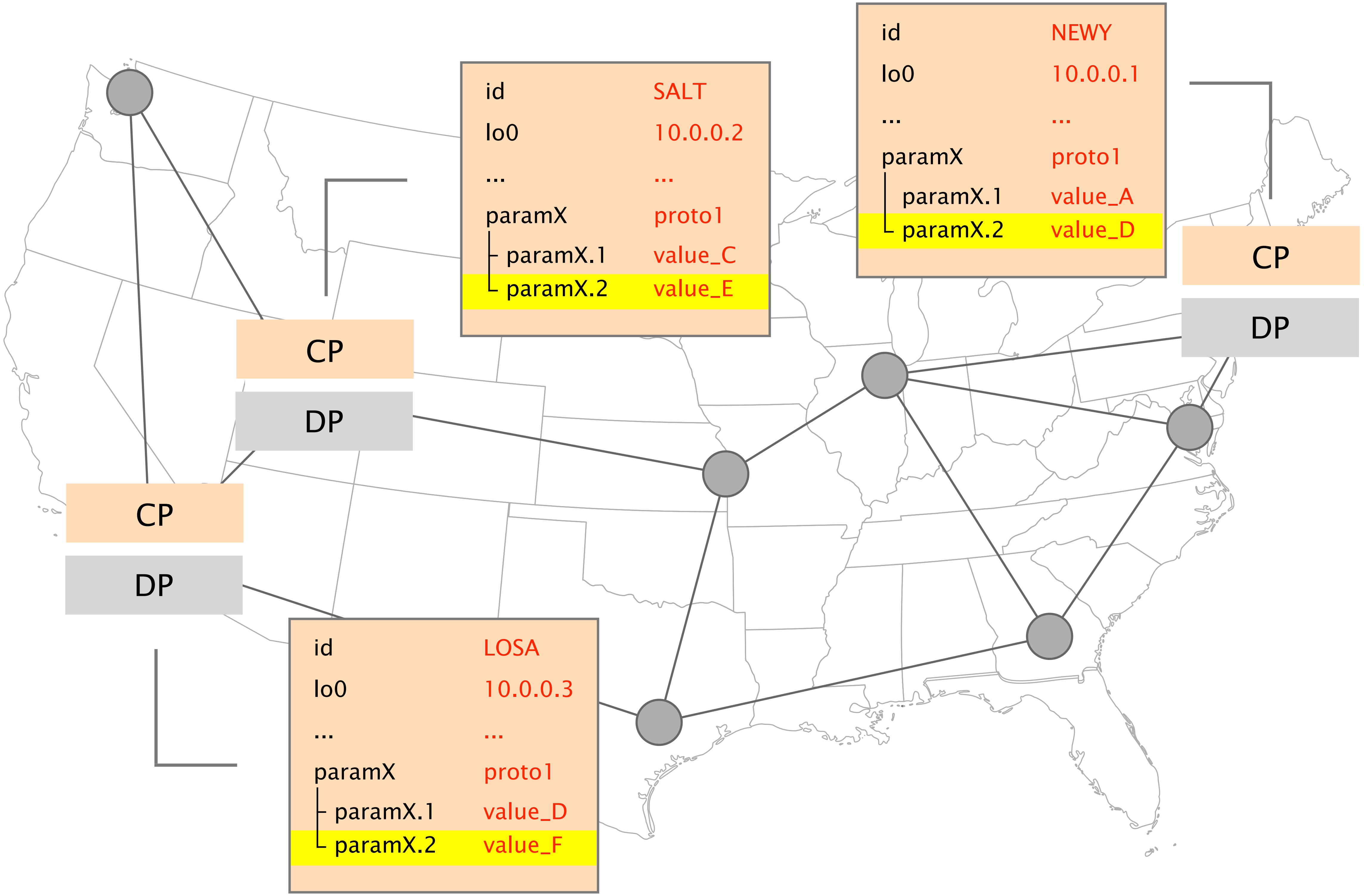


id	SALT
lo0	10.0.0.2
...	...
paramX	proto1
├ paramX.1	value_C
└ paramX.2	value_E

id	NEWY
lo0	10.0.0.1
...	...
paramX	proto1
├ paramX.1	value_A
└ paramX.2	value_D

id	LOSA
lo0	10.0.0.3
...	...
paramX	proto1
├ paramX.1	value_D
└ paramX.2	value_F

For the network to work properly, each parameter must be consistent network-wide

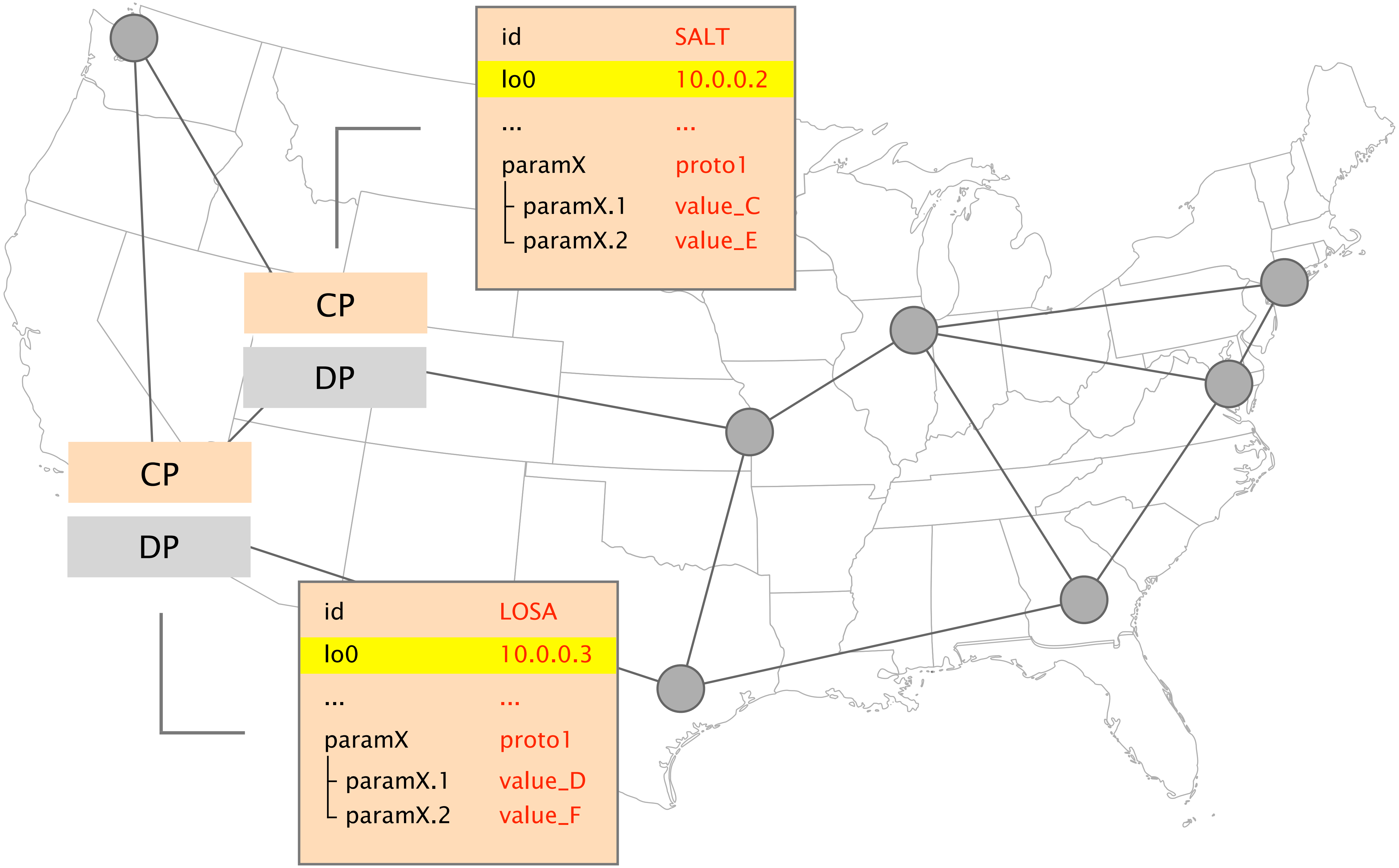


id	SALT
lo0	10.0.0.2
...	...
paramX	proto1
├ paramX.1	value_C
└ paramX.2	value_E

id	NEWY
lo0	10.0.0.1
...	...
paramX	proto1
├ paramX.1	value_A
└ paramX.2	value_D

id	LOSA
lo0	10.0.0.3
...	...
paramX	proto1
├ paramX.1	value_D
└ paramX.2	value_F

Reconfiguring the network consists in modifying some configuration parameters



CP

DP

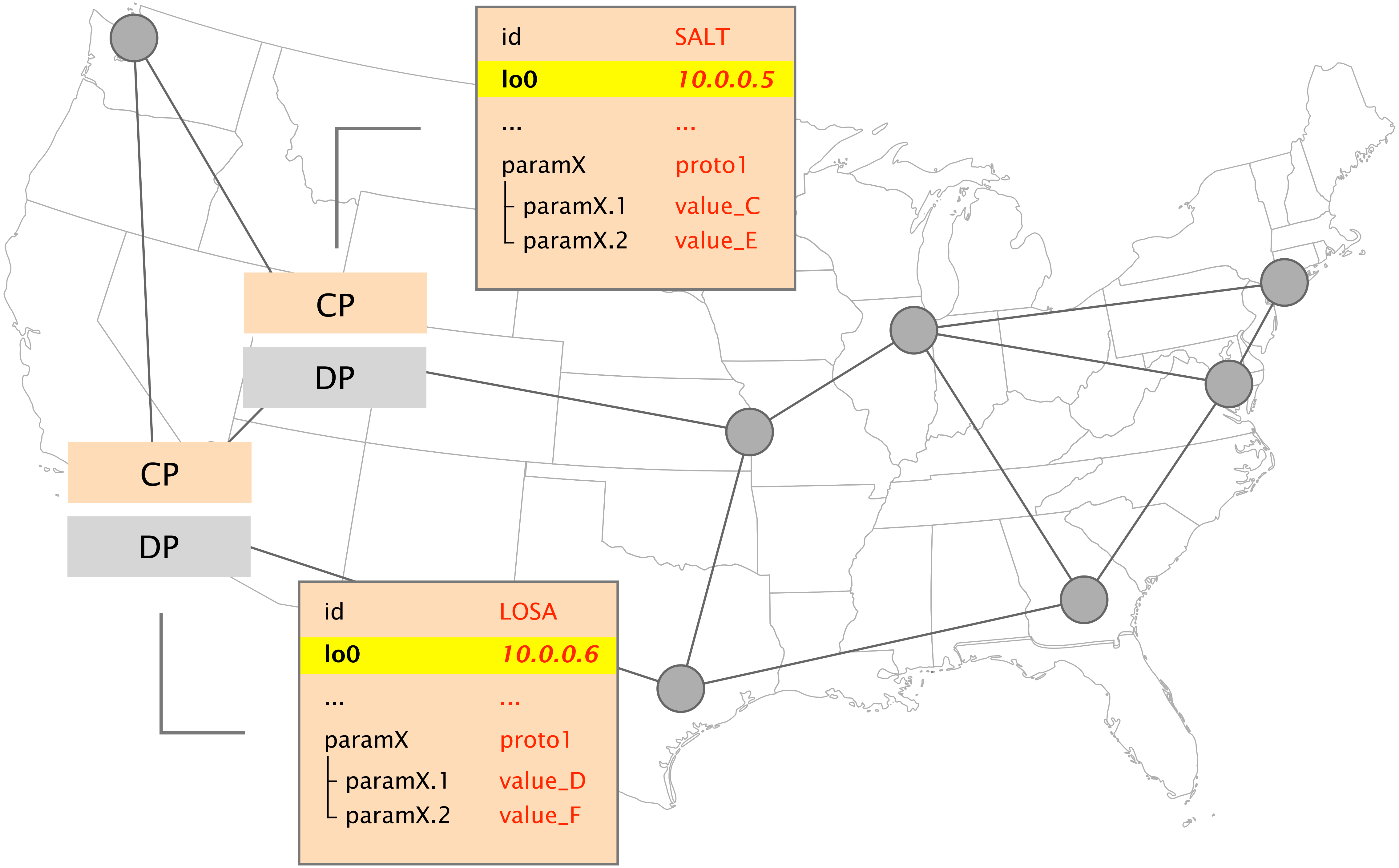
CP

DP

id	SALT
lo0	10.0.0.2
...	...
paramX	proto1
├ paramX.1	value_C
└ paramX.2	value_E

id	LOSA
lo0	10.0.0.3
...	...
paramX	proto1
├ paramX.1	value_D
└ paramX.2	value_F

Reconfiguring the network consists in modifying some configuration parameters



id	SALT
lo0	***10.0.0.5***
...	...
paramX	proto1
├ paramX.1	value_C
└ paramX.2	value_E

id	LOSA
lo0	***10.0.0.6***
...	...
paramX	proto1
├ paramX.1	value_D
└ paramX.2	value_F

Network reconfiguration is a day-to-day task

Configuring the network from scratch is done only once

Everything change after is a reconfiguration

Typical reconfigurations scenario include

- Updating the physical or logical infrastructure
- Managing resources (e.g., bandwidth, CPU, memory)
- Deploying new services

Network reconfiguration is hardly done right

Manually change a running network

device-by-device, using proprietary, low-level CLI interfaces

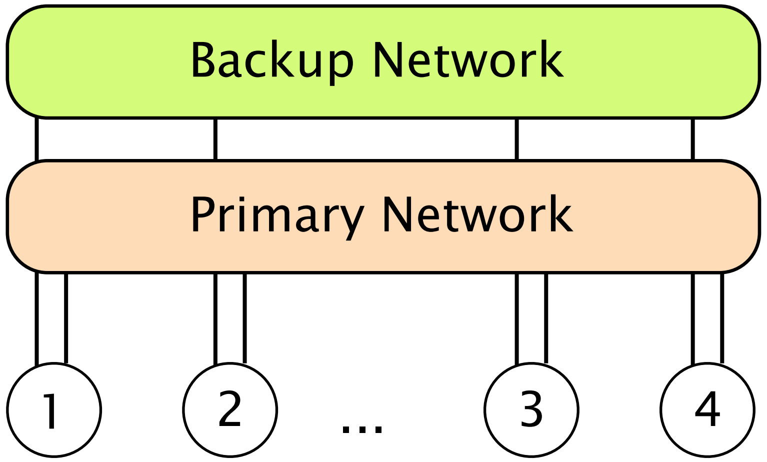
Ensuring consistency in every intermediate step

coordinating the changes across the entire network

Face routing and forwarding anomalies

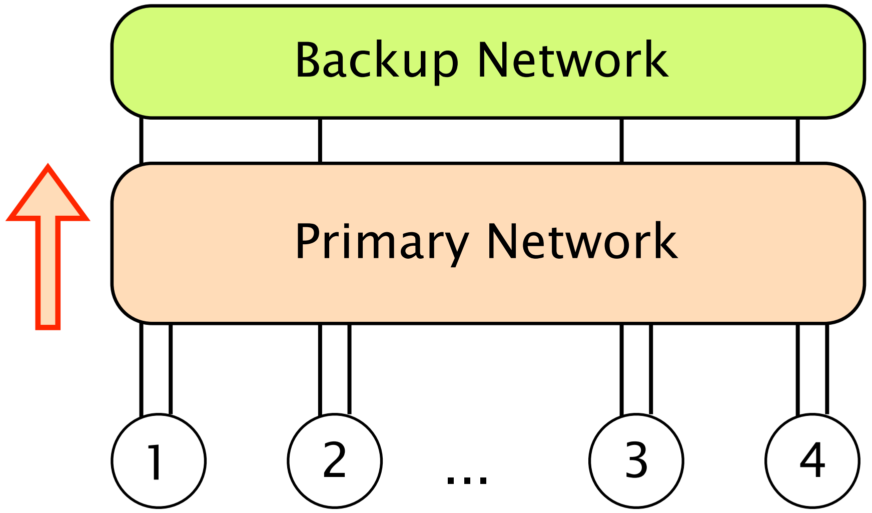
as non-reconfigured routers interact with reconfigured ones

At 12:47 AM PDT on April 21st, a network change was performed as part of our normal AWS scaling activities [...] The configuration change was to upgrade the capacity of the primary network.



Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region

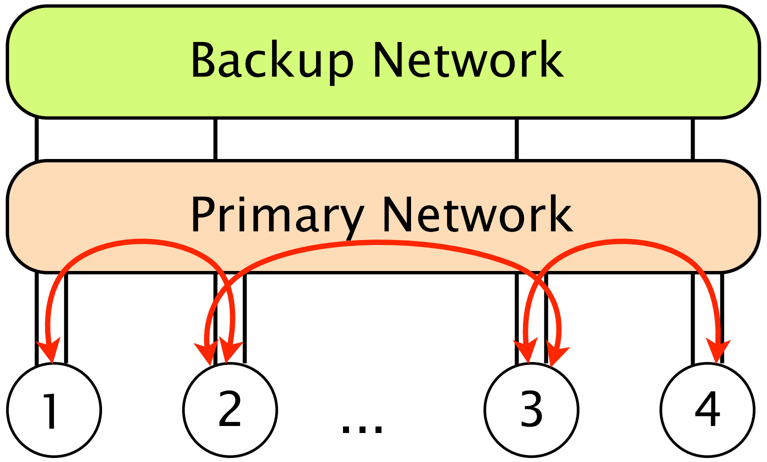
At 12:47 AM PDT on April 21st, a network change was performed as part of our normal AWS scaling activities [...]
The configuration change was to upgrade the capacity of the primary network.



Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region

At 12:47 AM PDT on April 21st, a network change was performed as part of our normal AWS scaling activities [...]. The configuration change was to upgrade the capacity of the primary network.

During the change, one of the standard steps is to shift traffic off of one of the redundant routers in the primary EBS network to allow the upgrade to happen.

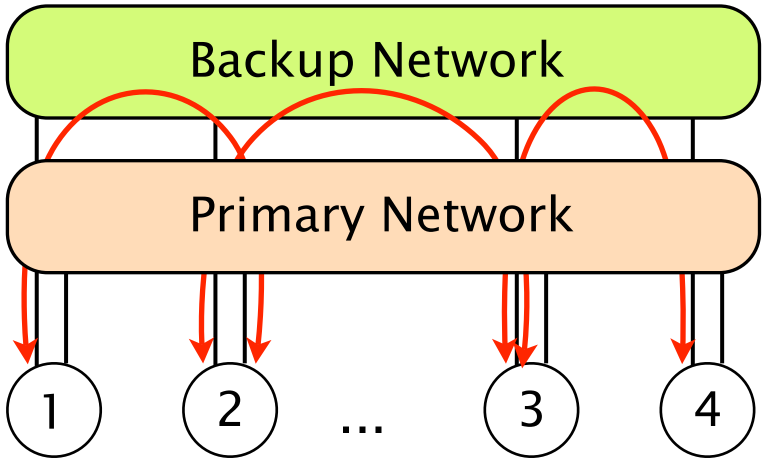


Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region

At 12:47 AM PDT on April 21st, a network change was performed as part of our normal AWS scaling activities [...]. The configuration change was to upgrade the capacity of the primary network.

During the change, one of the standard steps is to shift traffic off of one of the redundant routers in the primary EBS network to allow the upgrade to happen.

The traffic shift was executed incorrectly and rather than routing the traffic to the other router on the primary network, the traffic was routed onto the lower capacity redundant EBS network [...]



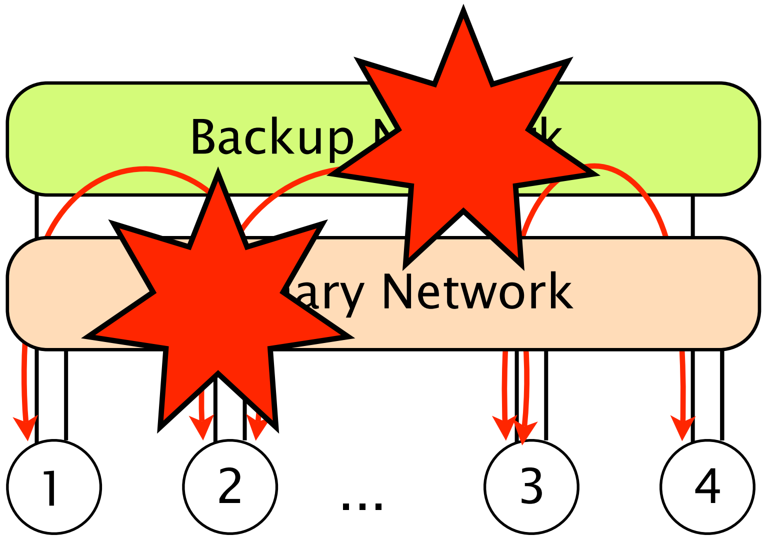
Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region

At 12:47 AM PDT on April 21st, a network change was performed as part of our normal AWS scaling activities [...]. The configuration change was to upgrade the capacity of the primary network.

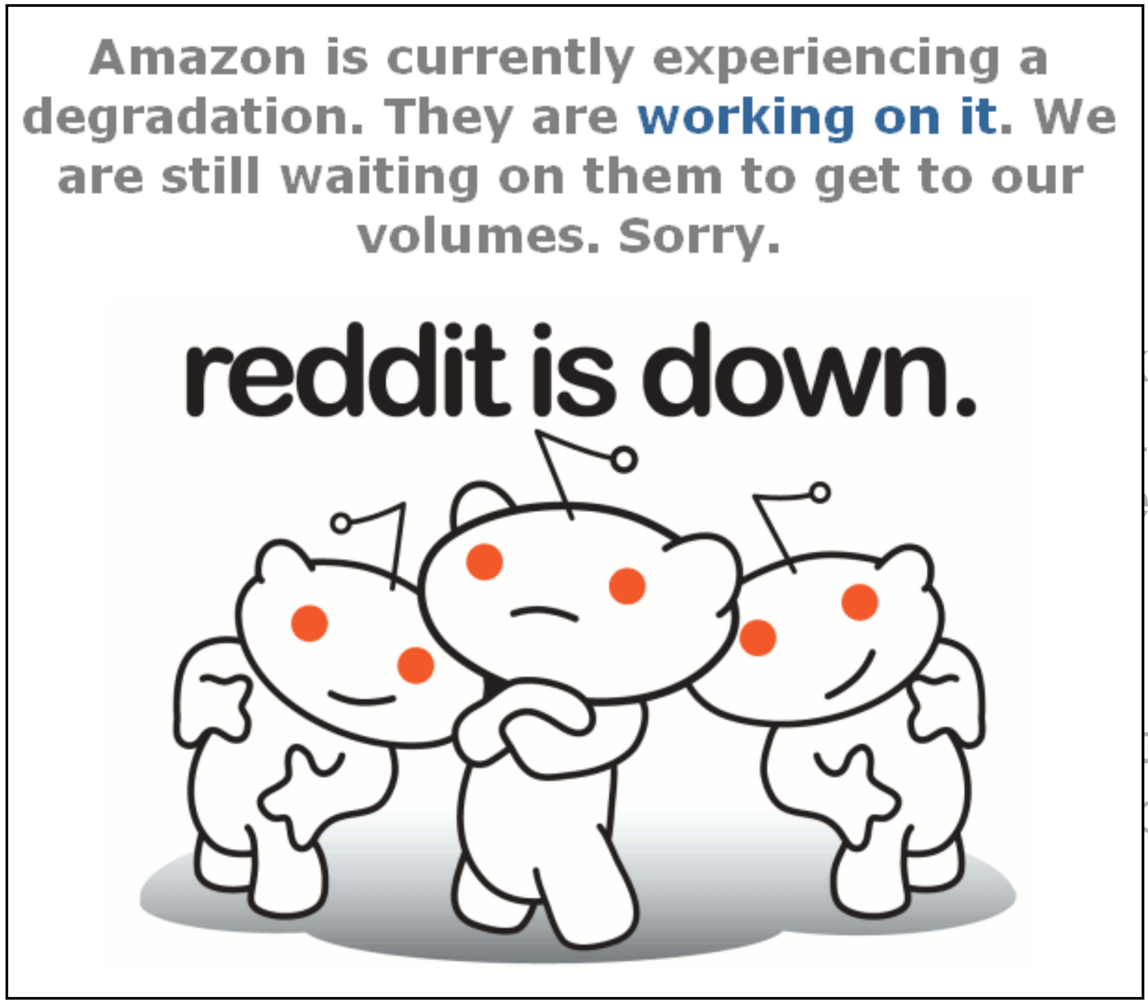
During the change, one of the standard steps is to shift traffic off of one of the redundant routers in the primary EBS network to allow the upgrade to happen.

The traffic shift was executed incorrectly and rather than routing the traffic to the other router on the primary network, the traffic was routed onto the lower capacity redundant EBS network [...]

Unlike a normal network interruption, this change disconnected both the primary and secondary network simultaneously, leaving the affected nodes completely isolated from one another.



Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region



Amazon is currently experiencing a degradation. They are working on it. We are still waiting on them to get to our volumes. Sorry.

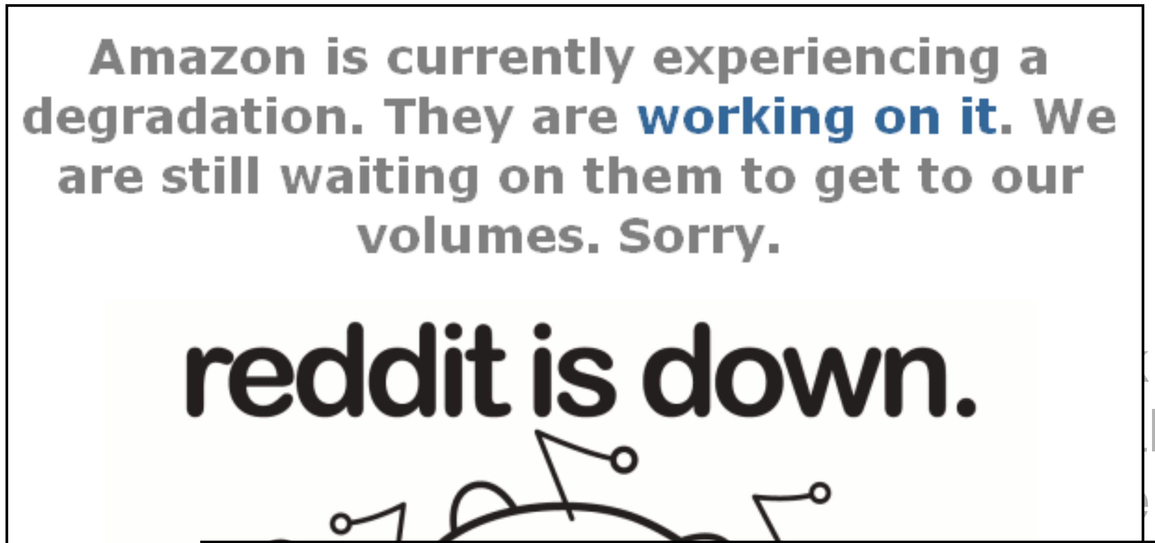
change was
ling activities [...].
the capacity of the

eps is to shift traffic
primary EBS network

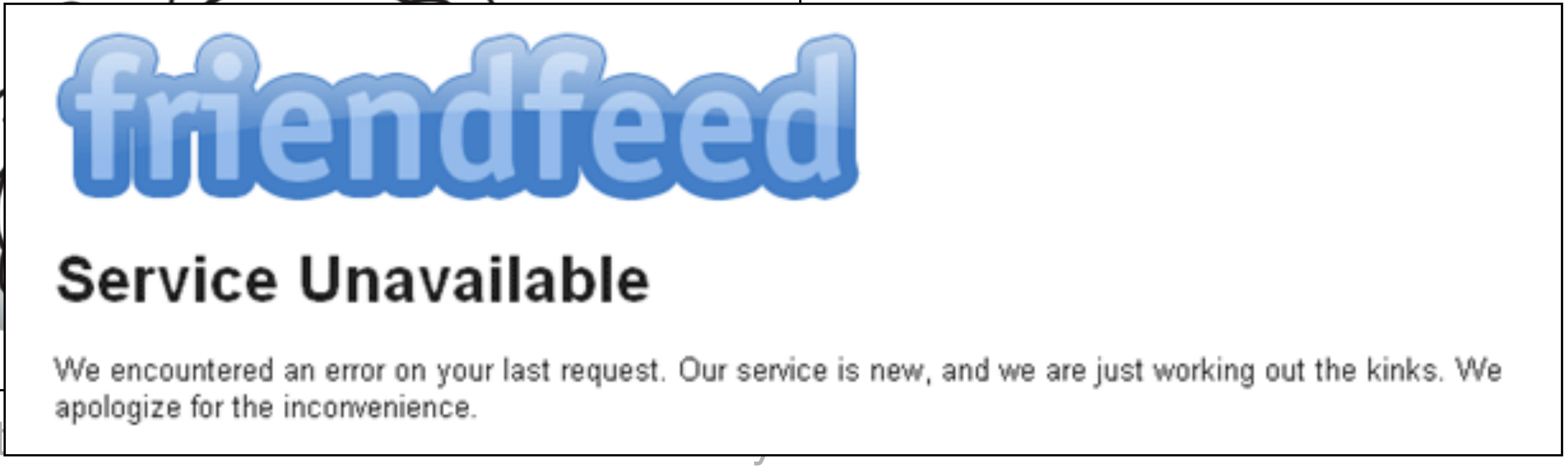
The traffic shift was executed incorrectly and rather than routing the traffic to the other router on the primary network, the traffic was routed onto the lower capacity redundant EBS network [...]

Unlike a normal network interruption, this change disconnected both the primary and secondary network simultaneously, leaving the affected nodes completely isolated from one another.

Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region



Amazon is currently experiencing a degradation. They are working on it. We are still waiting on them to get to our volumes. Sorry.



Service Unavailable

We encountered an error on your last request. Our service is new, and we are just working out the kinks. We apologize for the inconvenience.

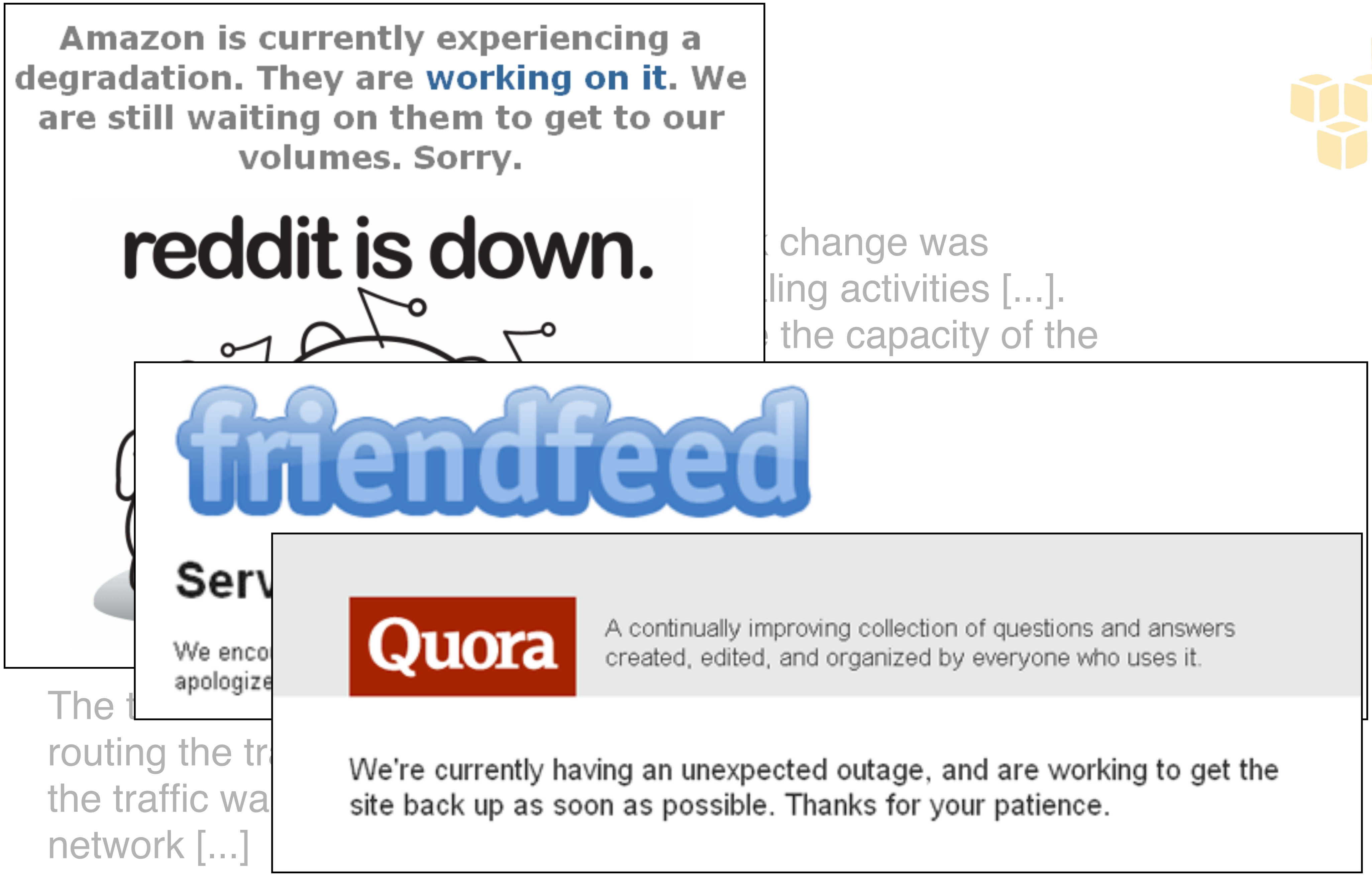
change was
ling activities [...].
the capacity of the

The t
routing the traffic to the other router on the primary network, the traffic was routed onto the lower capacity redundant EBS network [...]

Unlike a normal network interruption, this change disconnected both the primary and secondary network simultaneously, leaving the affected nodes completely isolated from one another.

Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region

amazon webservices™



Amazon is currently experiencing a degradation. They are working on it. We are still waiting on them to get to our volumes. Sorry.

reddit is down.

friendfeed

Serv

We enco
apologize

Quora

A continually improving collection of questions and answers created, edited, and organized by everyone who uses it.

We're currently having an unexpected outage, and are working to get the site back up as soon as possible. Thanks for your patience.

change was
ling activities [...].
the capacity of the

The t
routing the tr
the traffic wa
network [...]

Unlike a normal network interruption, this change disconnected both the primary and secondary network simultaneously, leaving the affected nodes completely isolated from one another.

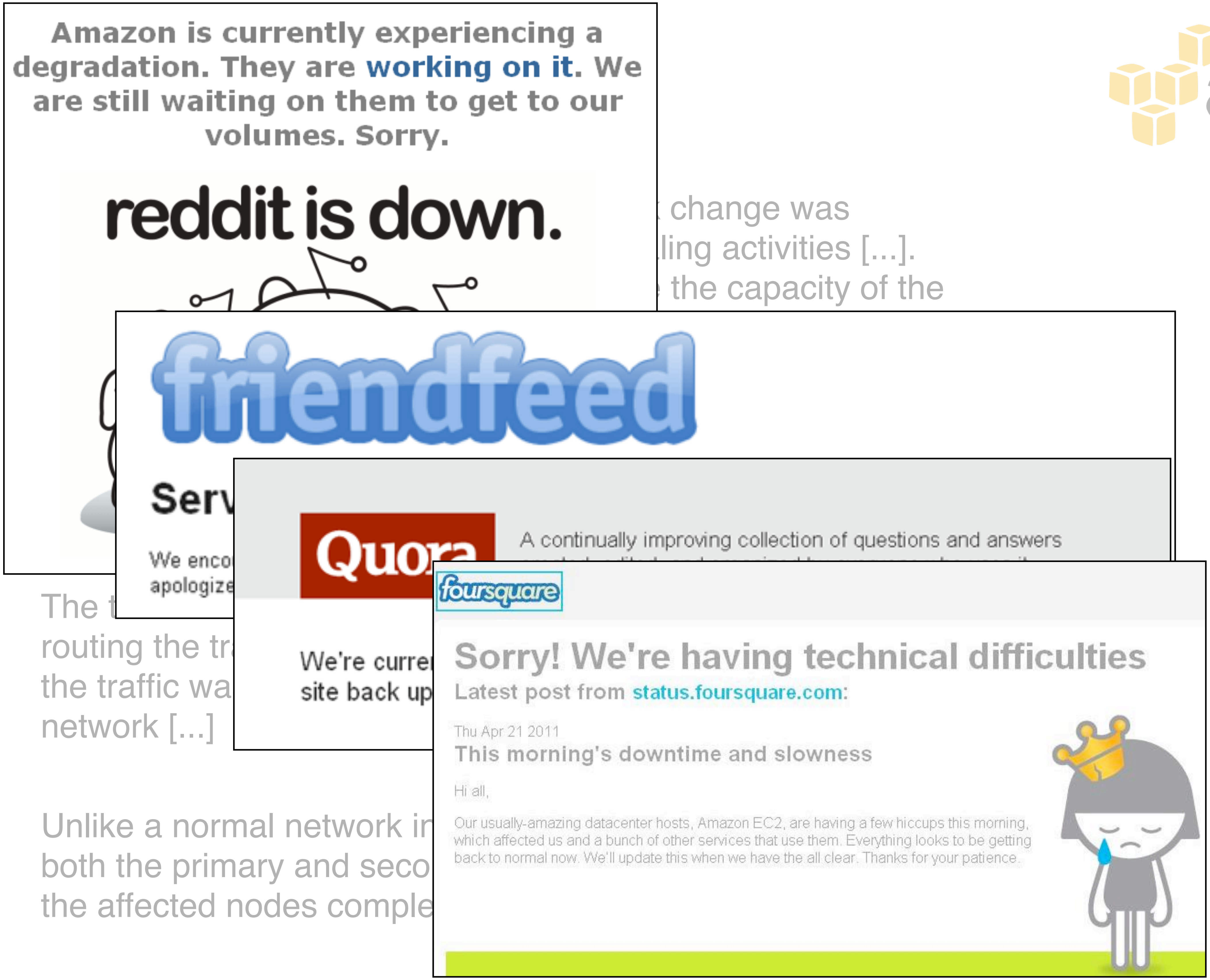
Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region

amazon
webservices™

change was
ling activities [...].
the capacity of the

The t
routing the tr
the traffic wa
network [...]

Unlike a normal network in
both the primary and seco
the affected nodes comple



Amazon is currently experiencing a degradation. They are working on it. We are still waiting on them to get to our volumes. Sorry.

reddit is down.

friendfeed

Serv

We enco
apologize

Quora

A continually improving collection of questions and answers

We're curre
site back up

foursquare

Sorry! We're having technical difficulties

Latest post from status.foursquare.com:

Thu Apr 21 2011

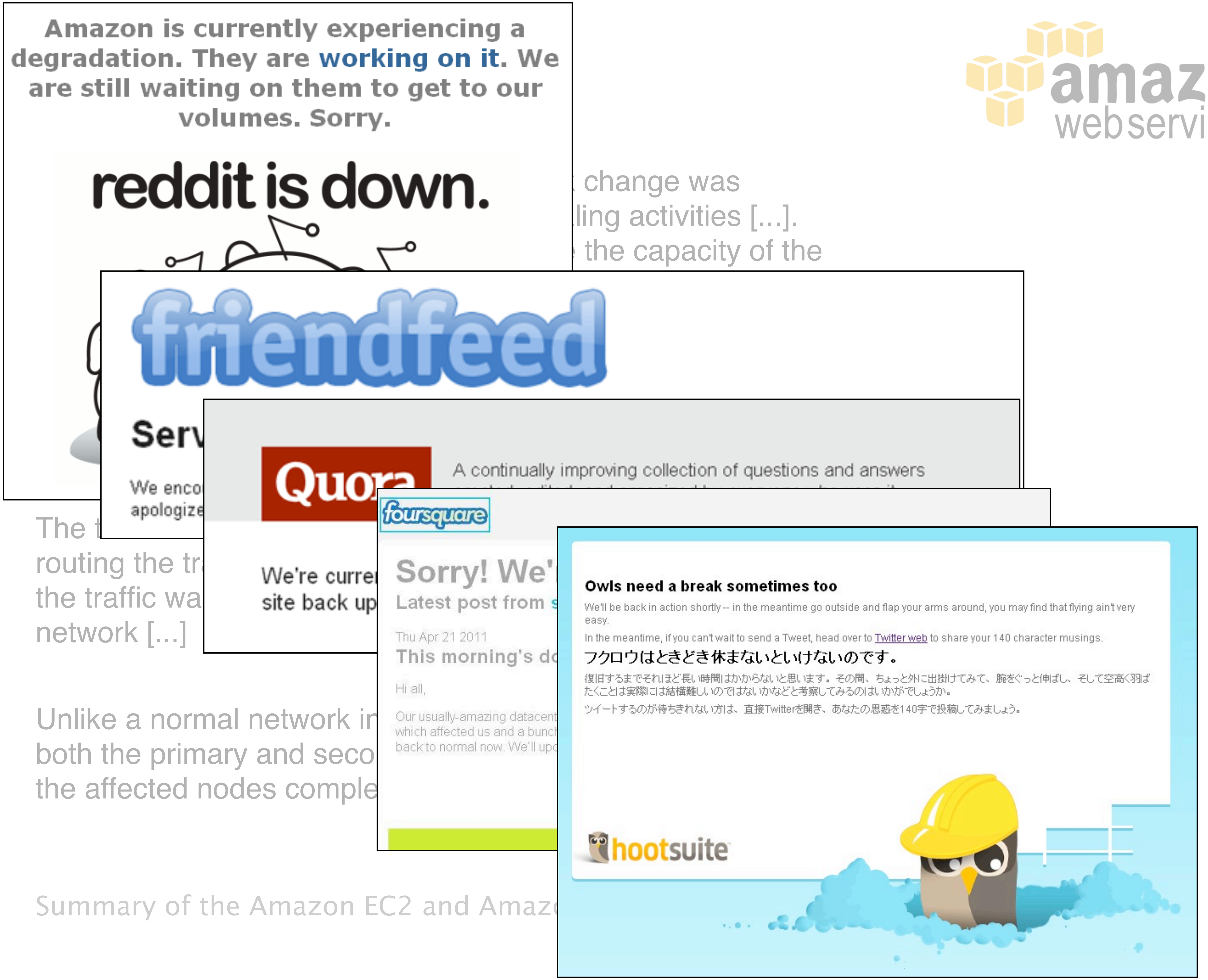
This morning's downtime and slowness

Hi all,

Our usually-amazing datacenter hosts, Amazon EC2, are having a few hiccups this morning, which affected us and a bunch of other services that use them. Everything looks to be getting back to normal now. We'll update this when we have the all clear. Thanks for your patience.

Summary of the Amazon EC2 and Amazon RDS Service Disruption in the US East Region

change was
ling activities [...].
the capacity of the



Amazon is currently experiencing a degradation. They are working on it. We are still waiting on them to get to our volumes. Sorry.

reddit is down.

Serv

We enco
apologize

A continually improving collection of questions and answers

We're curre
site back up

Sorry! We'

Latest post from s

Thu Apr 21 2011

This morning's do

Hi all,

Our usually-amazing datacent
which affected us and a bunch
back to normal now. We'll upd

Owls need a break sometimes too

We'll be back in action shortly -- in the meantime go outside and flap your arms around, you may find that flying ain't very easy.

In the meantime, if you can't wait to send a Tweet, head over to Twitter web to share your 140 character musings.

フクロウはときどき休まないといけないのです。

復旧するまでそれほど長い時間はかからないと思います。その間、ちょっと外に出掛けてみて、腕をぐっと伸ばし、そして空高く羽ばたくことは実際には結構難しいのではないかなどと考察してみるのはいかがでしょうか。

ツイートするのが待ちきれない方は、直接Twitterを開き、あなたの思惑を140字で投稿してみましょう。

The t
routing the tr
the traffic wa
network [...]

Unlike a normal network in
both the primary and seco
the affected nodes comple

Summary of the Amazon EC2 and Amaz

Amazon is currently experiencing a degradation. They are working on it. We are still waiting on them to get to our volumes. Sorry.

change was
ling activities [...].
the capacity of the

The trigger for this event was a poorly executed network reconfiguration

the traffic wa
network [...]

site back up

Latest post from s

Thu Apr 21 2011

This morning's do

Hi all,

Our usually-amazing datacent
which affected us and a bunch
back to normal now. We'll upd

We'll be back in action shortly -- in the meantime go outside and flap your arms around, you may find that flying ain't very easy.

In the meantime, if you can't wait to send a Tweet, head over to Twitter web to share your 140 character musings.

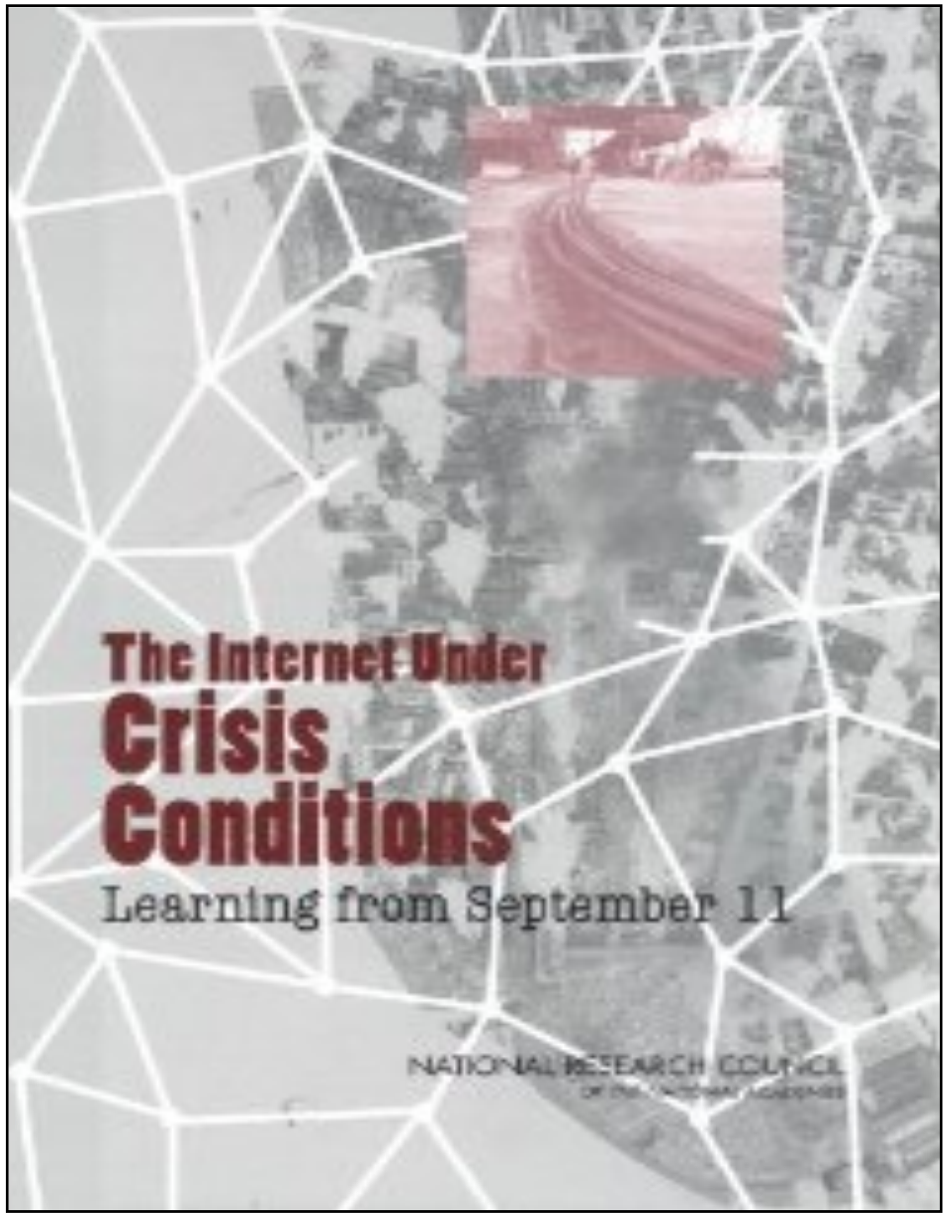
フクロウはときどき休まないといけないのです。

復旧するまでそれほど長い時間はかからないと思います。その間、ちょっと外に出掛けてみて、腕をぐっと伸ばし、そして空高く羽ばたくことは実際には結構難しいのではないかなどと考察してみるのはいかがでしょうか。

ツイートするのが待ちきれない方は、直接Twitterを開き、あなたの思惑を140字で投稿してみましょう。

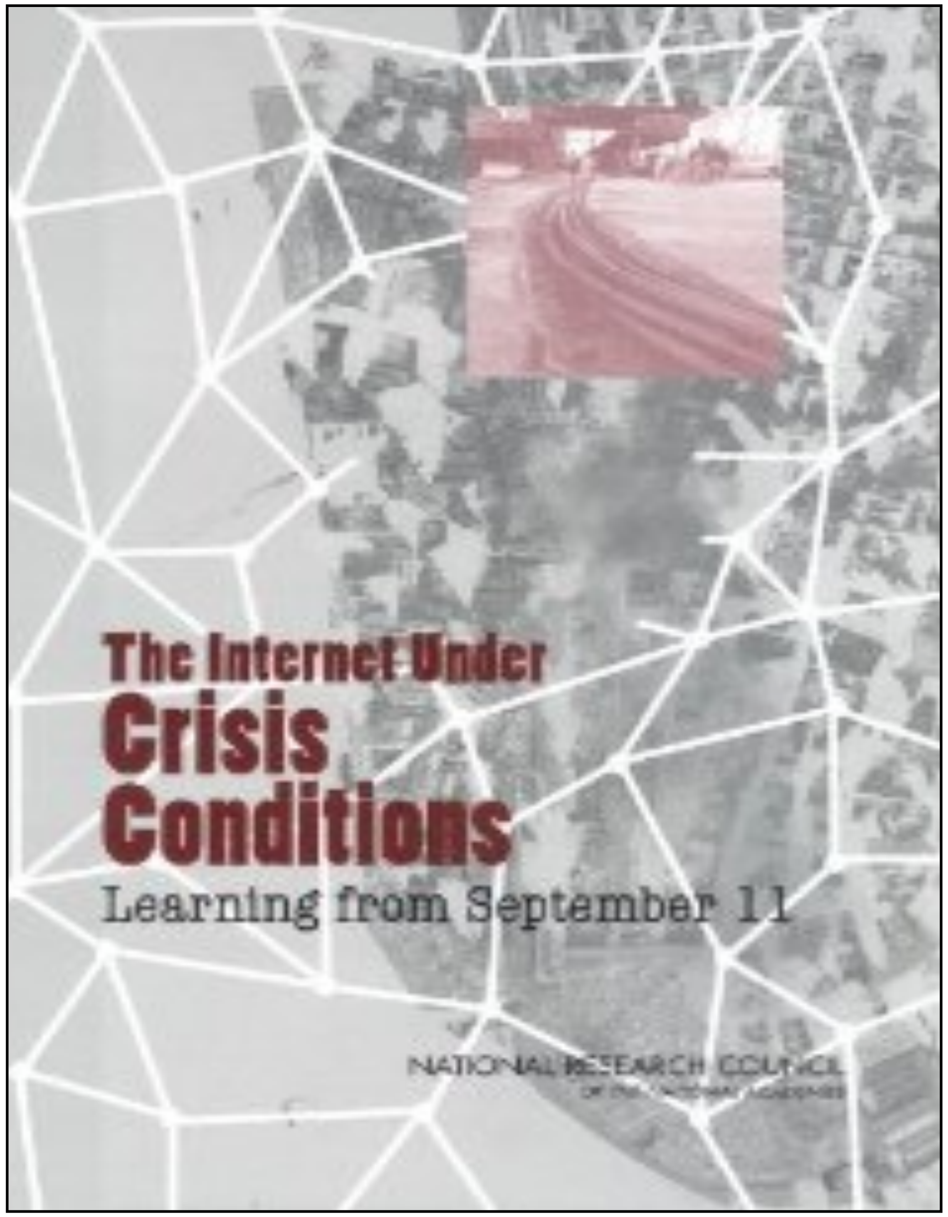
Unlike a normal network in
both the primary and seco
the affected nodes comple

Summary of the Amazon EC2 and Amazo



The rate of BGP routing advertisements suggests that the Internet was more stable than normal on September 11

National Research Council. The Internet Under Crisis Conditions: Learning from September 11



The rate of BGP routing advertisements suggests that the Internet was more stable than normal on September 11

Information from network operators suggests that many operators were watching the news instead of making normal changes to their routers.

National Research Council. The Internet Under Crisis Conditions: Learning from September 11

Our ultimate goal is to enable anomaly-free routing reconfiguration

Progressively reconfigure a running network
without creating any anomaly

Our approach mixes theory and practice

Develop reconfiguration techniques which are

- provably correct
- efficient
- automatic
- backward compatible

Methods and Techniques for Disruption-free Network Reconfiguration

Background

What is a network?

2 Intradomain reconfiguration

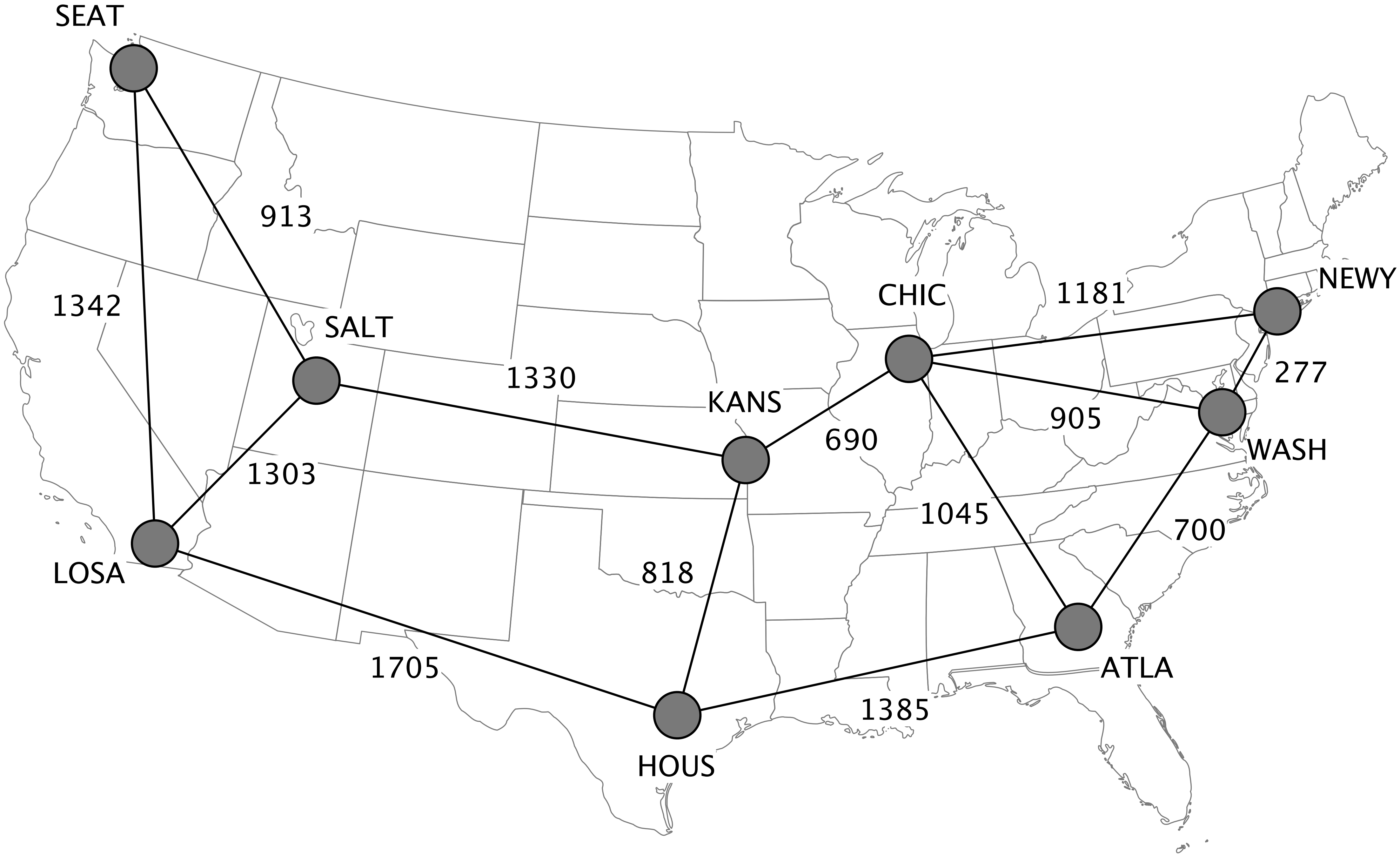
Find a reconfiguration ordering

Interdomain reconfiguration

Overcome inherent complexity

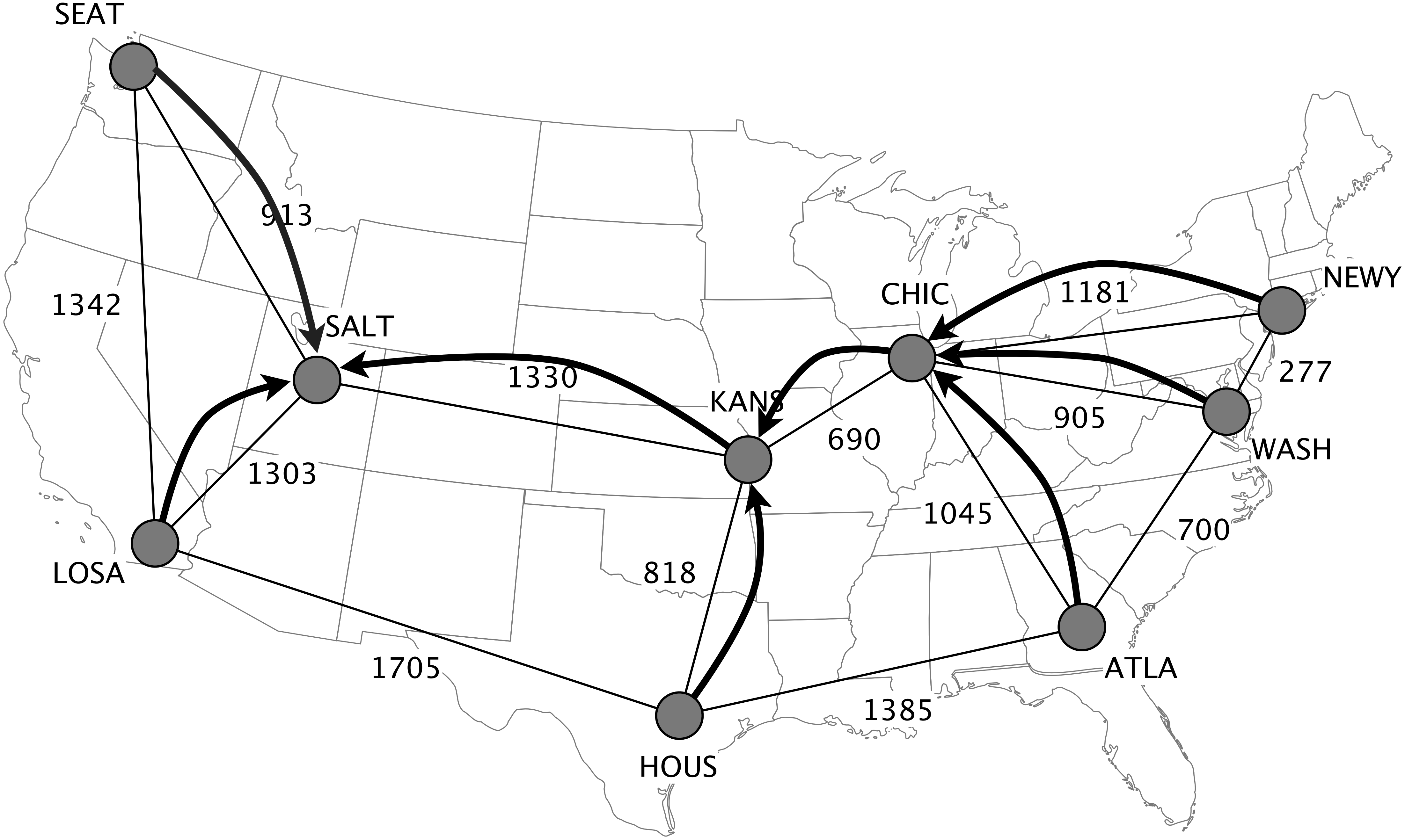
Intradomain routing protocols (IGP) rule traffic forwarding within a routing domain

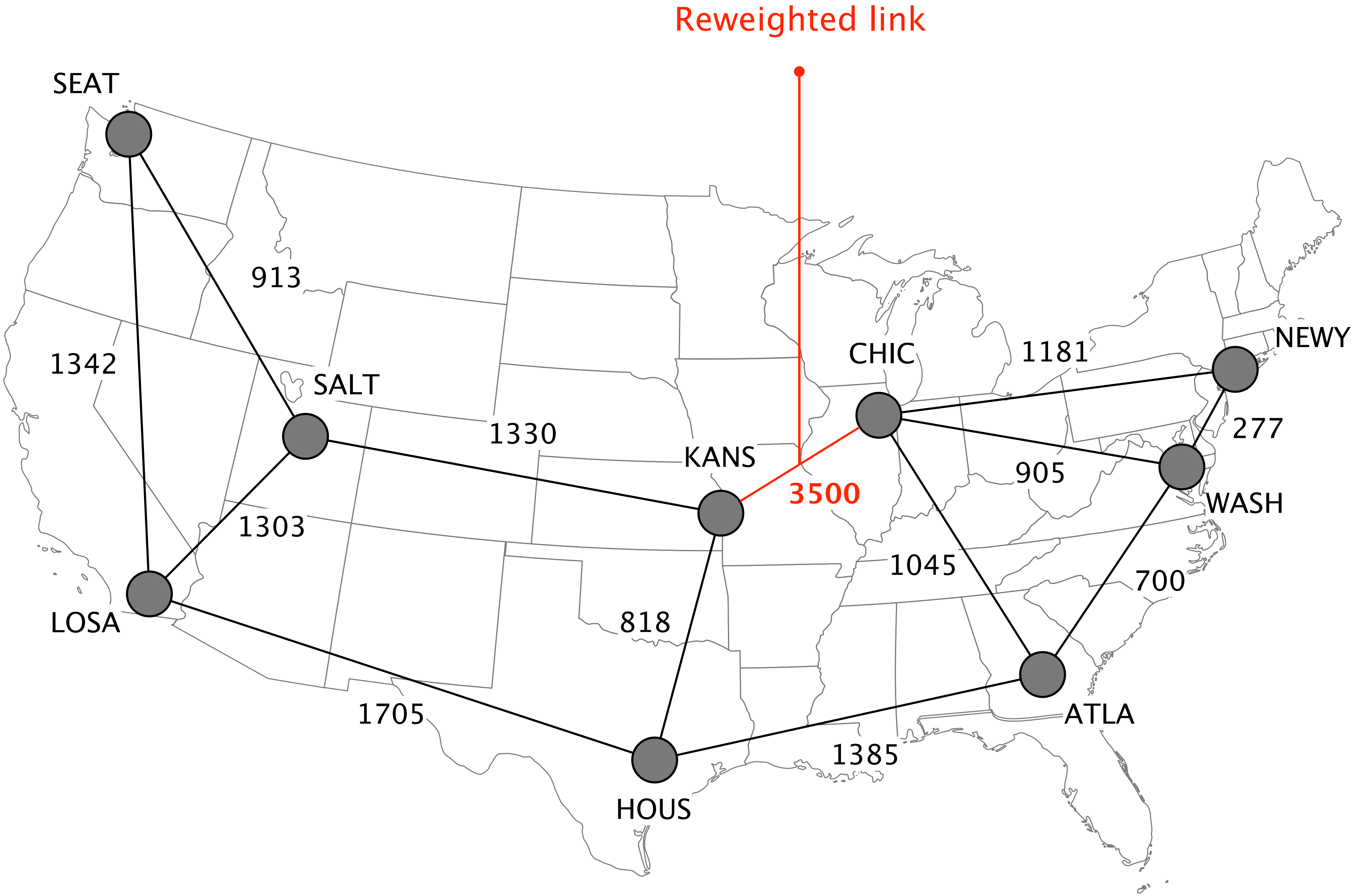
The US research network (Abilene, Internet2)



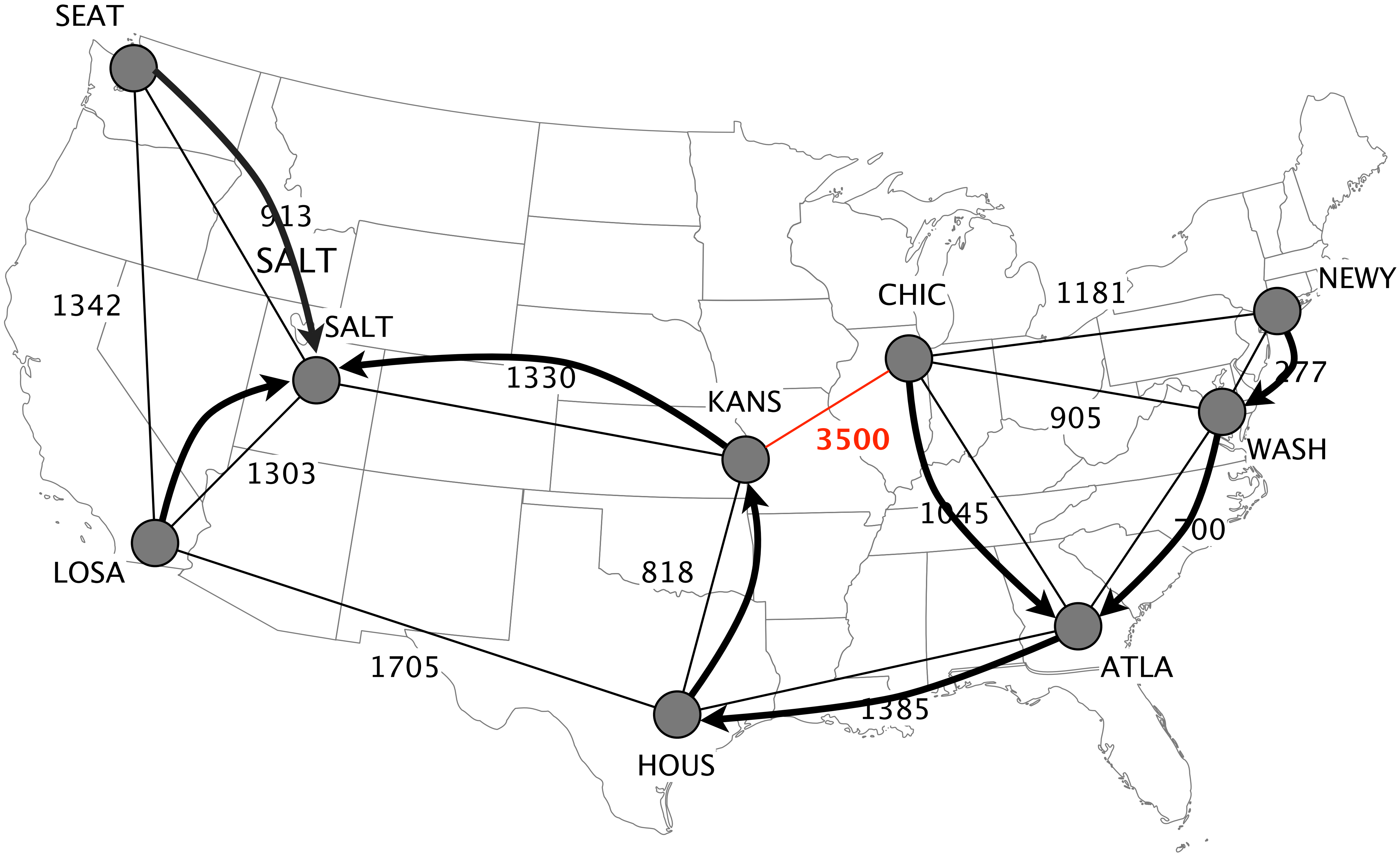
IGP enables each router to compute the shortest path to reach every other router

Forwarding paths towards SALT



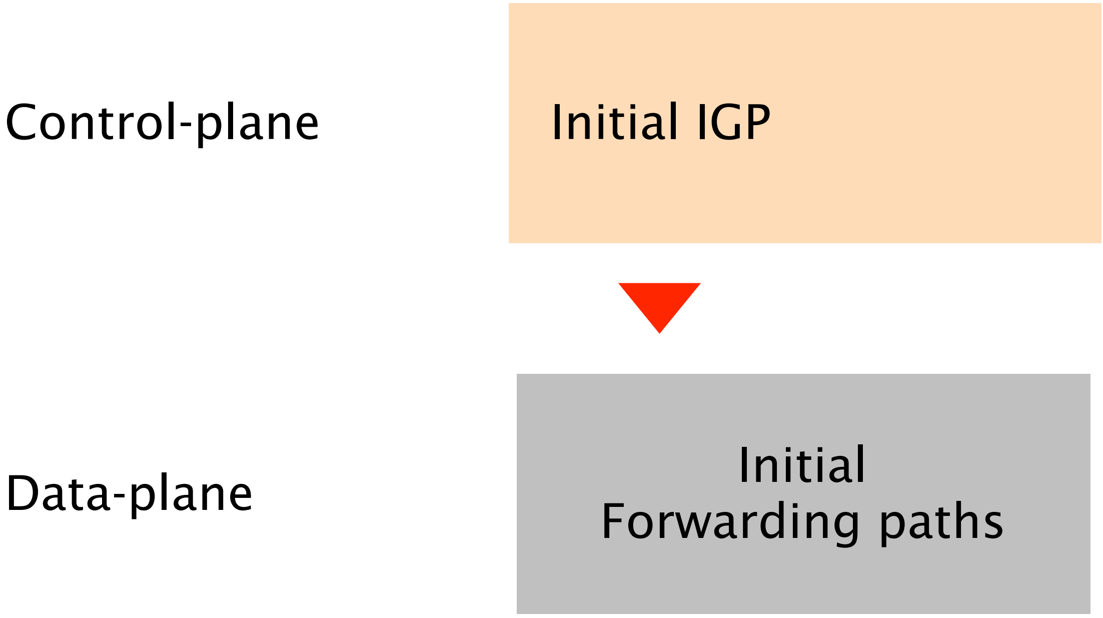

Reweighted link
SEAT
913
1342
SALT
1330
KANS
3500
CHIC
1181
NEWY
277
905
WASH
1303
LOSA
818
1045
700
1705
1385
ATLA
HOUS

Final forwarding paths towards SALT



Reconfiguring the IGP usually requires running two routing planes (*)

Abstract model of a router



Control-plane

Initial IGP

Data-plane

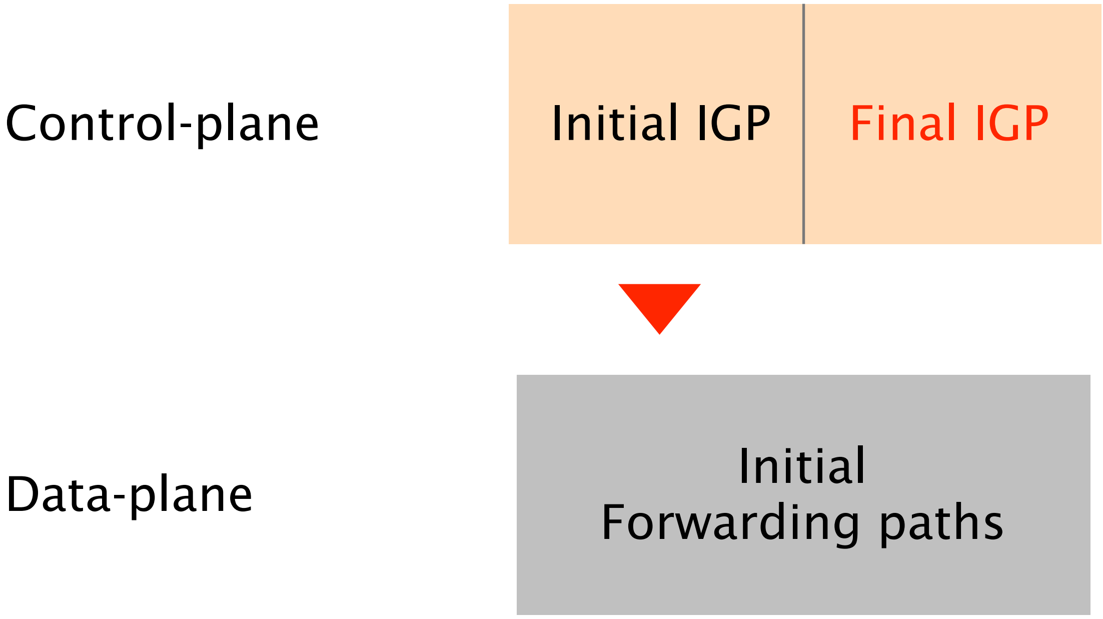
Initial Forwarding paths

At first, the initial IGP dictates the forwarding paths being used

(*) [Gill03, Pepelnjak07, Herrero10, Smith12]

Reconfiguring the IGP usually requires running two routing planes (*)

Abstract model of a router



Control-plane

Initial IGP

Final IGP

Data-plane

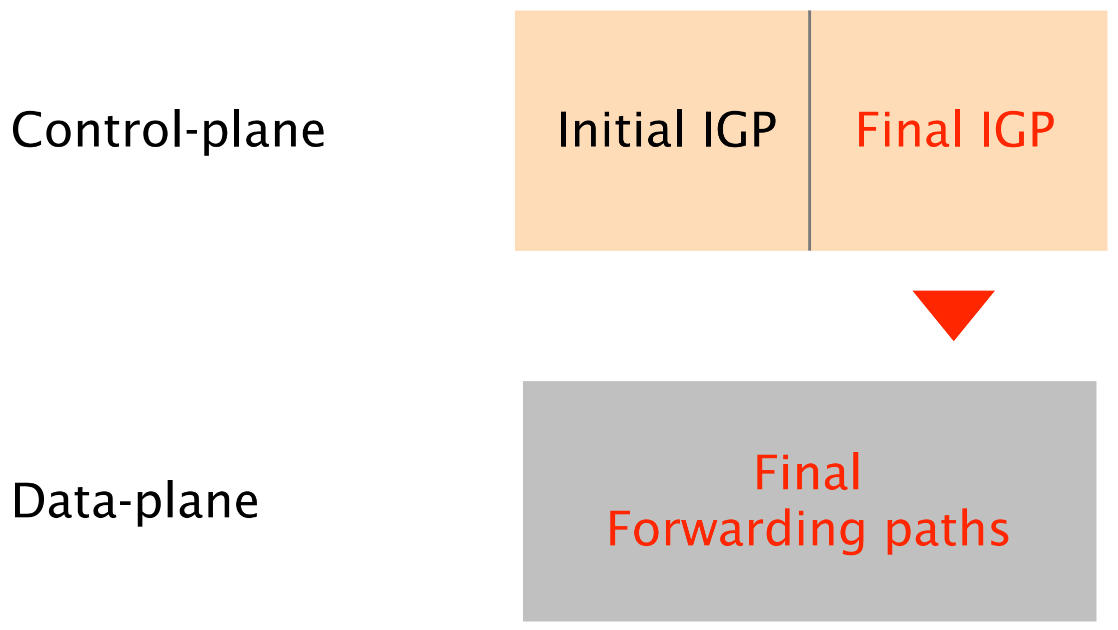
Initial
Forwarding paths

Then, the final IGP is introduced without changing the forwarding

(*) [Gill03, Pepelnjak07, Herrero10, Smith12]

Reconfiguring the IGP usually requires running two routing planes (*)

Abstract model of a router

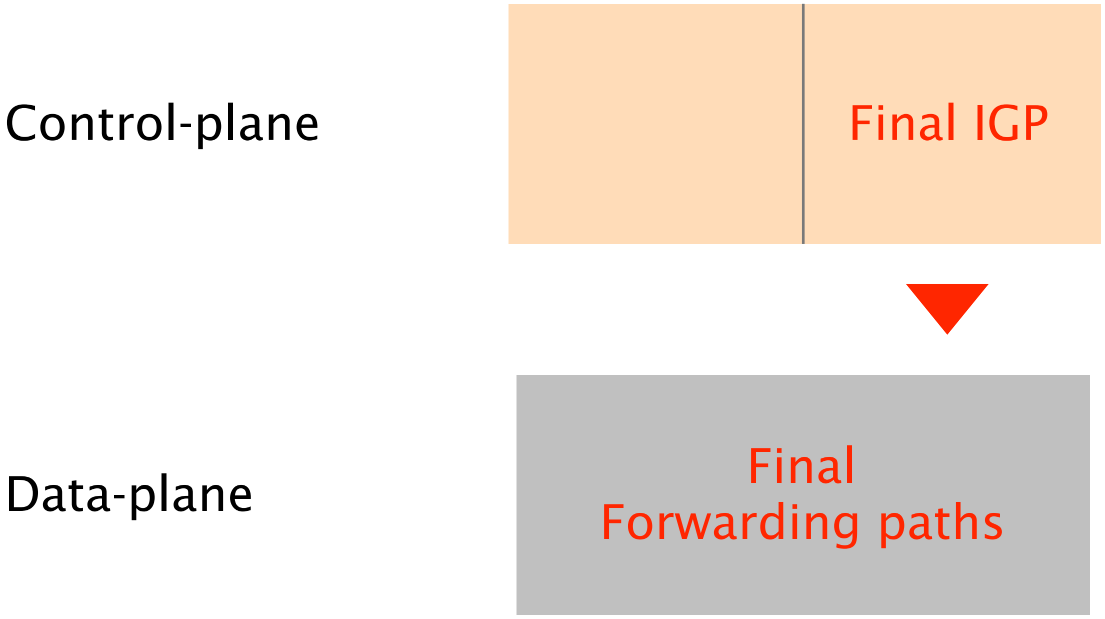


After having converged,
the final IGP is activated by
flipping the preference

(*) [Gill03, Pepelnjak07, Herrero10, Smith12]

Reconfiguring the IGP usually requires running two routing planes (*)

Abstract model of a router



Control-plane

Final IGP

Data-plane

Final
Forwarding paths

After having converged,
the final IGP is activated by
flipping the preference

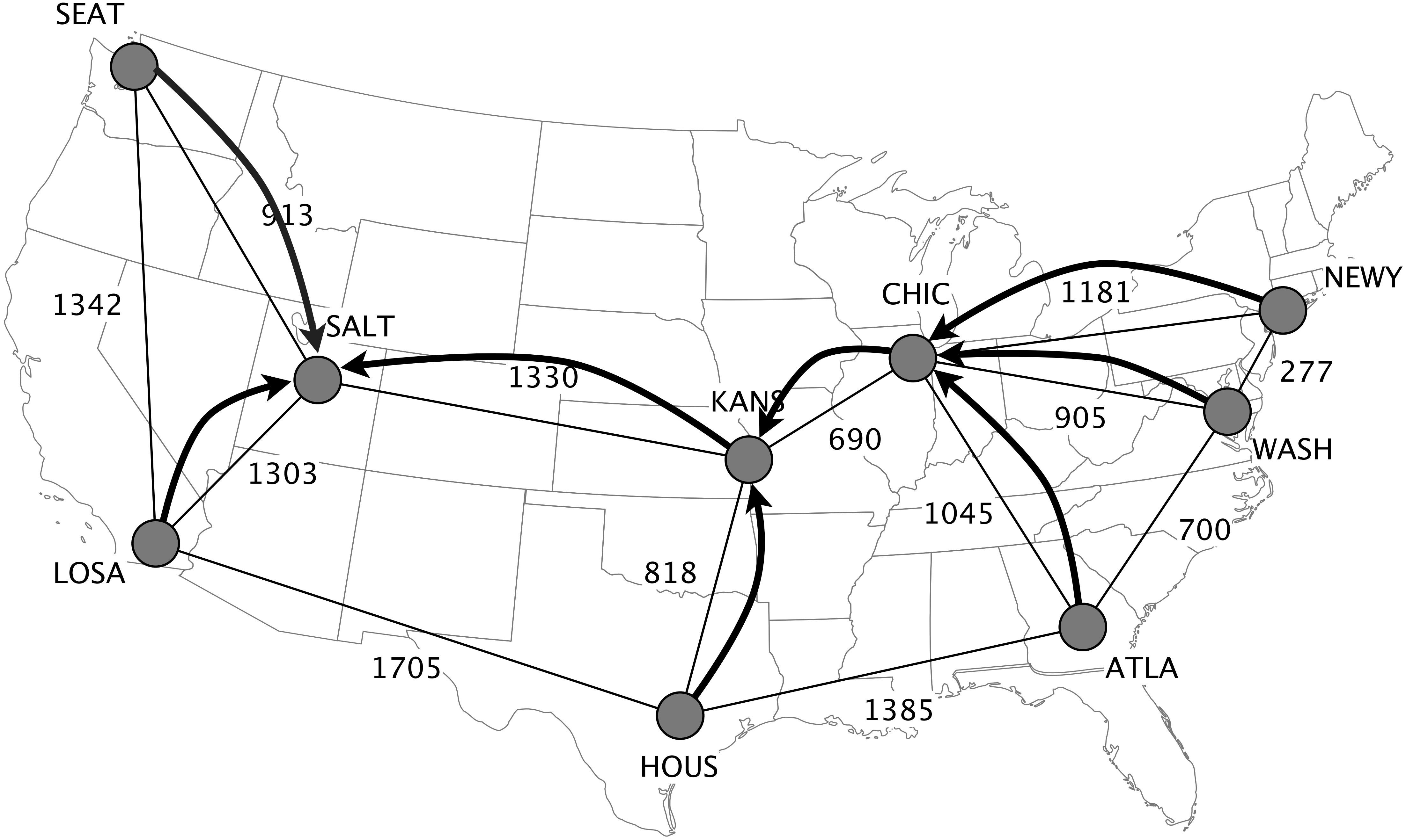
MIGRATED

(*) [Gill03, Pepelnjak07, Herrero10, Smith12]

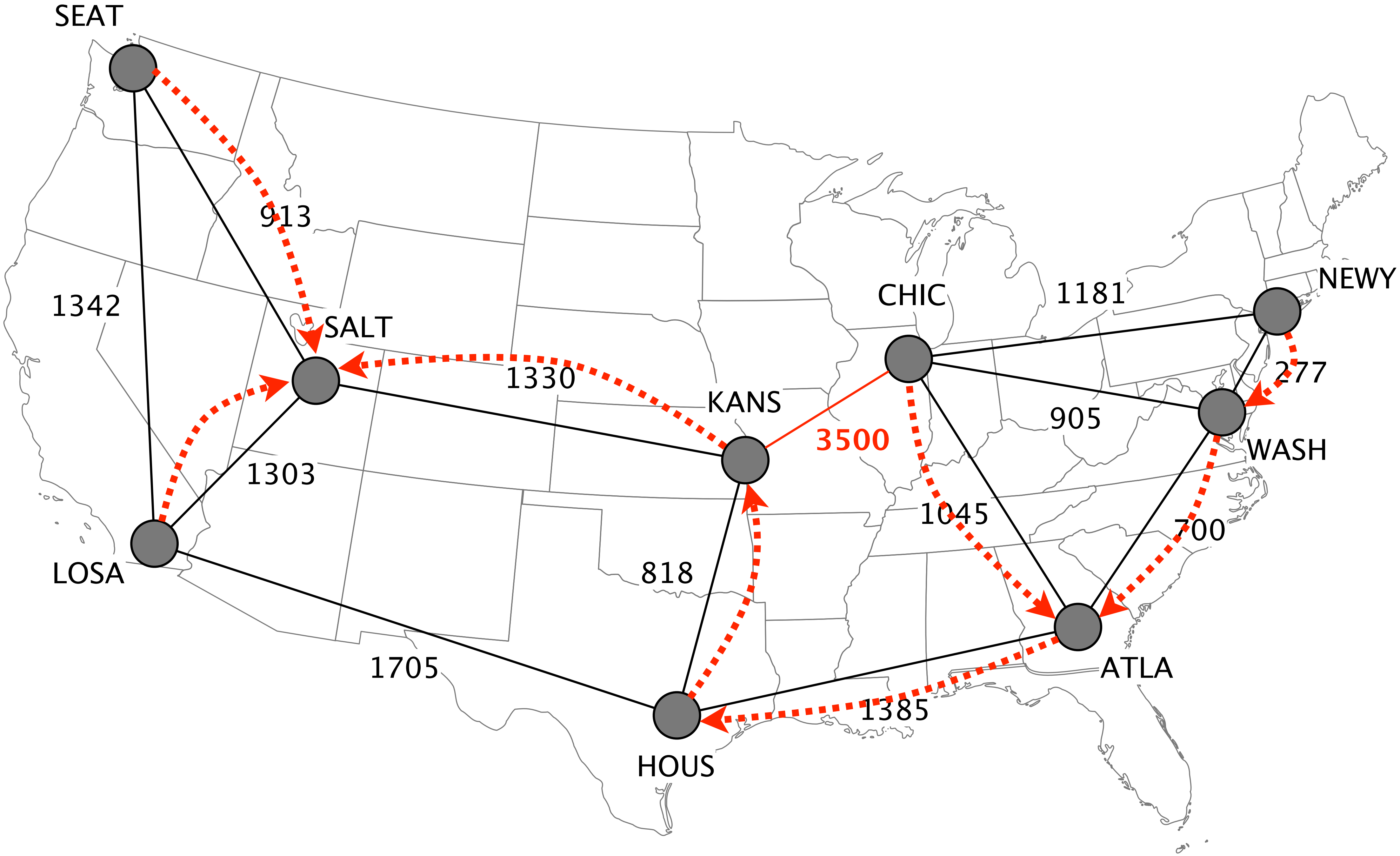
problem

Find an ordering in which to activate the final IGP without causing any forwarding anomalies

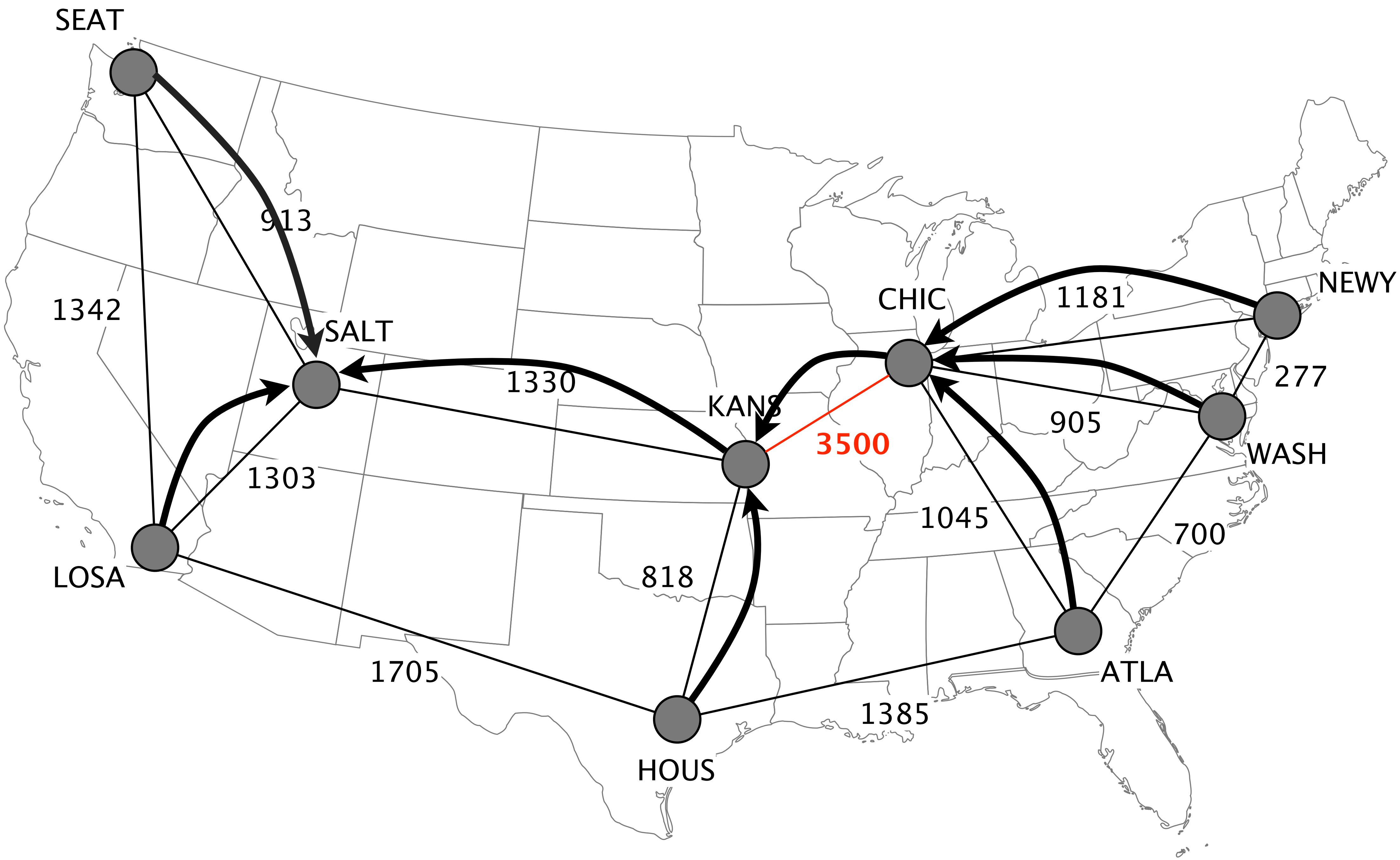
Initial forwarding paths towards SALT



Final forwarding paths towards SALT

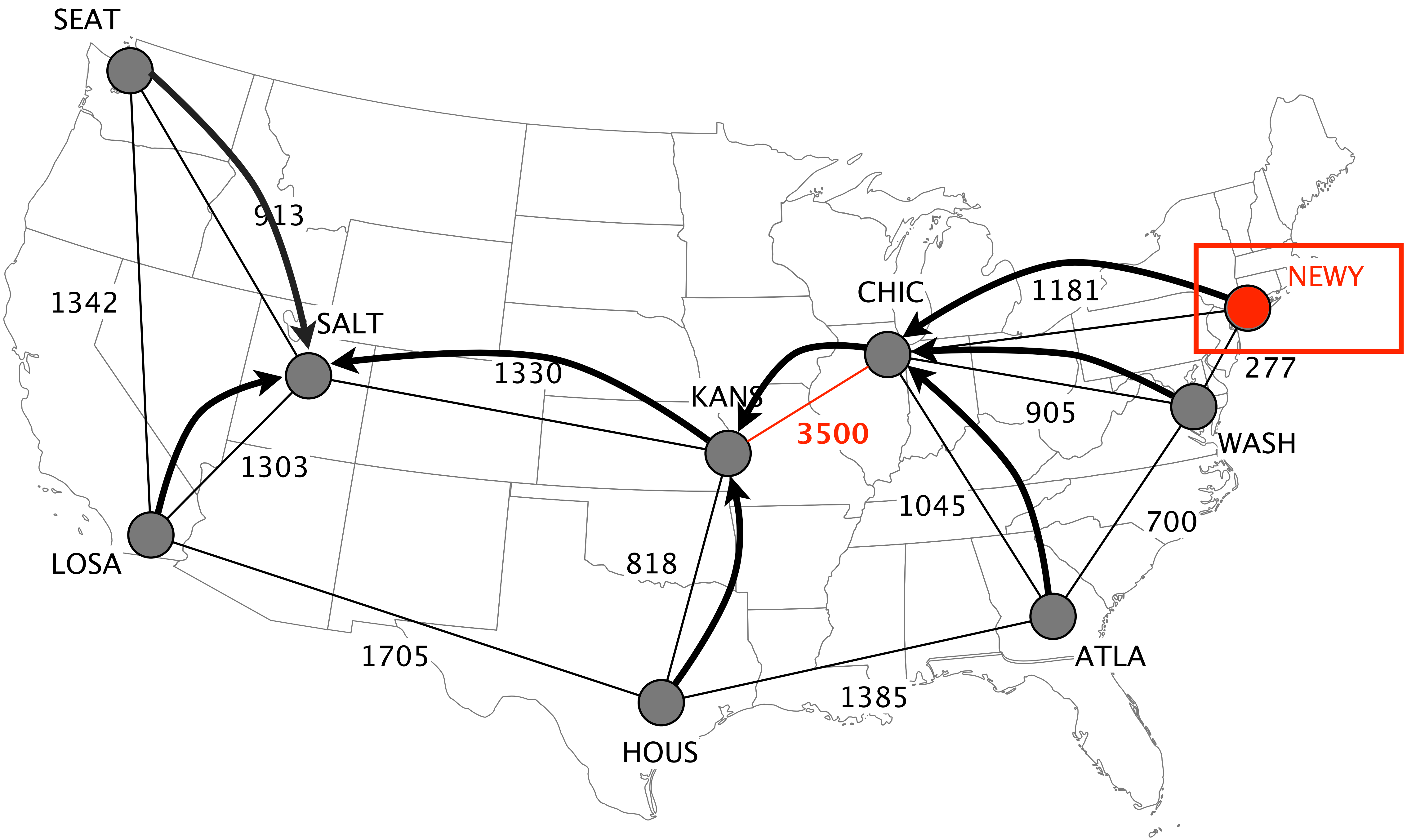


Migrated	[]
To migrate	[NEWY, WASH, CHIC, ATLA, ...]



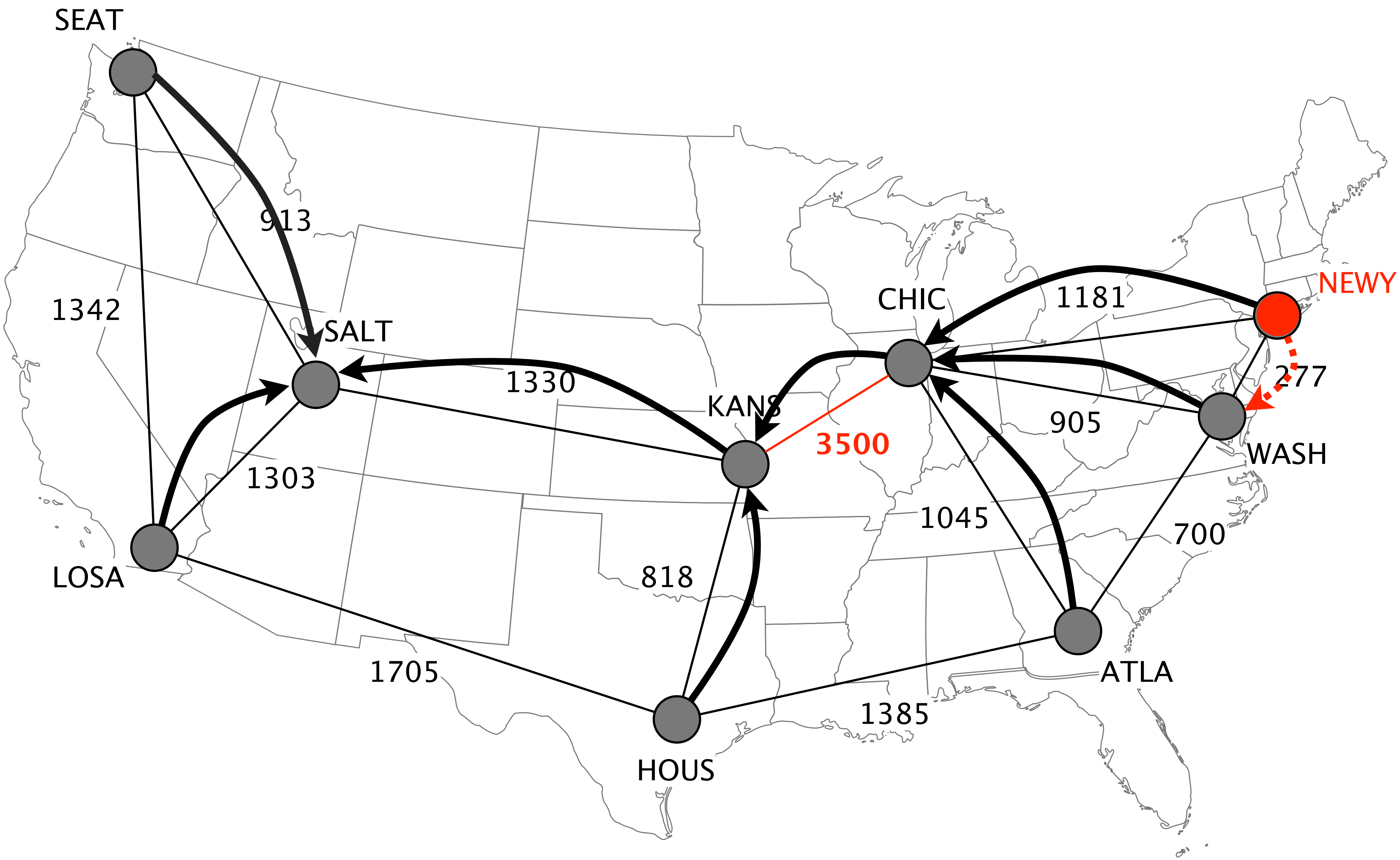
Migrated []

To migrate [NEWY, WASH, CHIC, ATLA, ...]



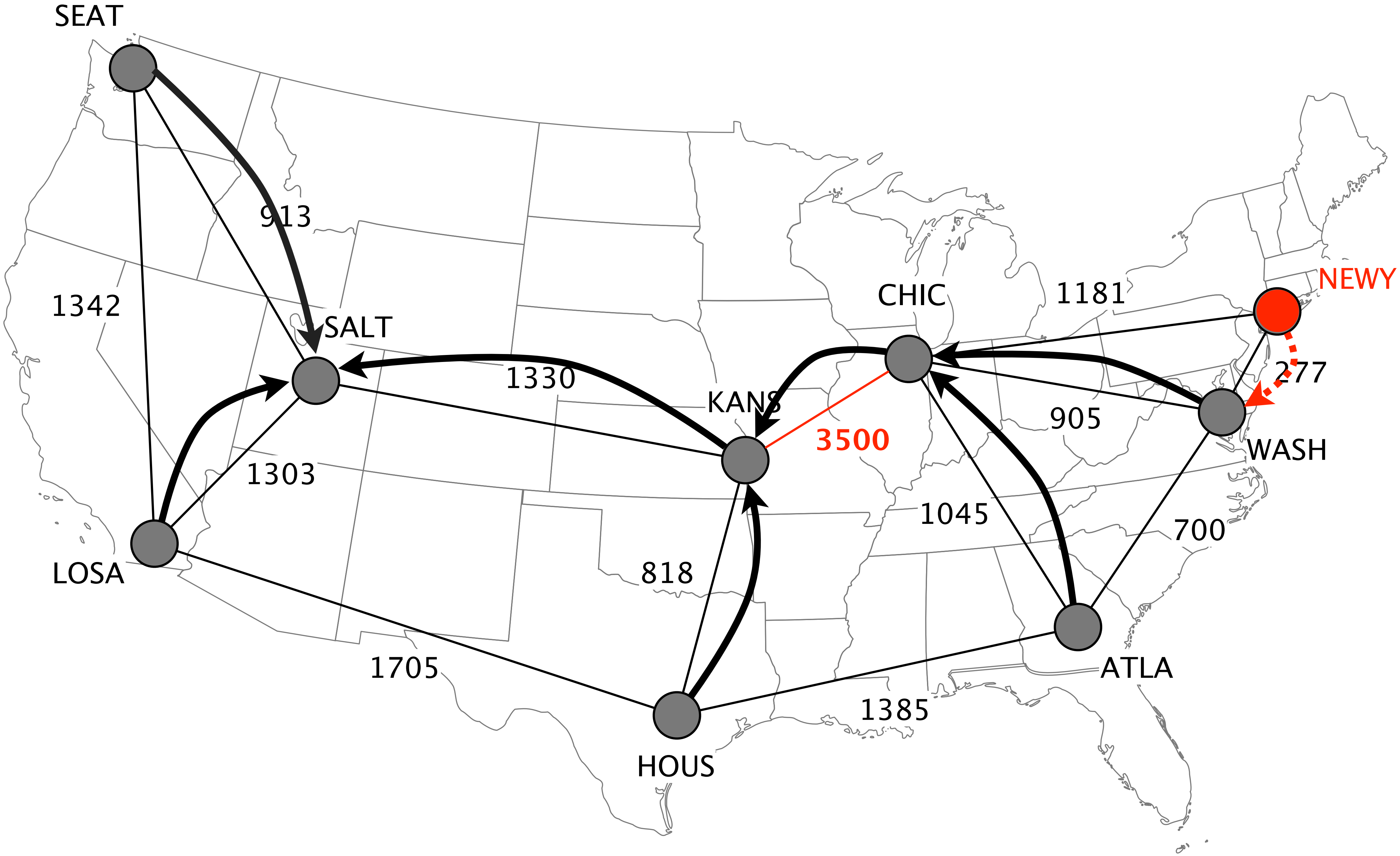
Migrated []

To migrate [NEWY, WASH, CHIC, ATLA, ...]



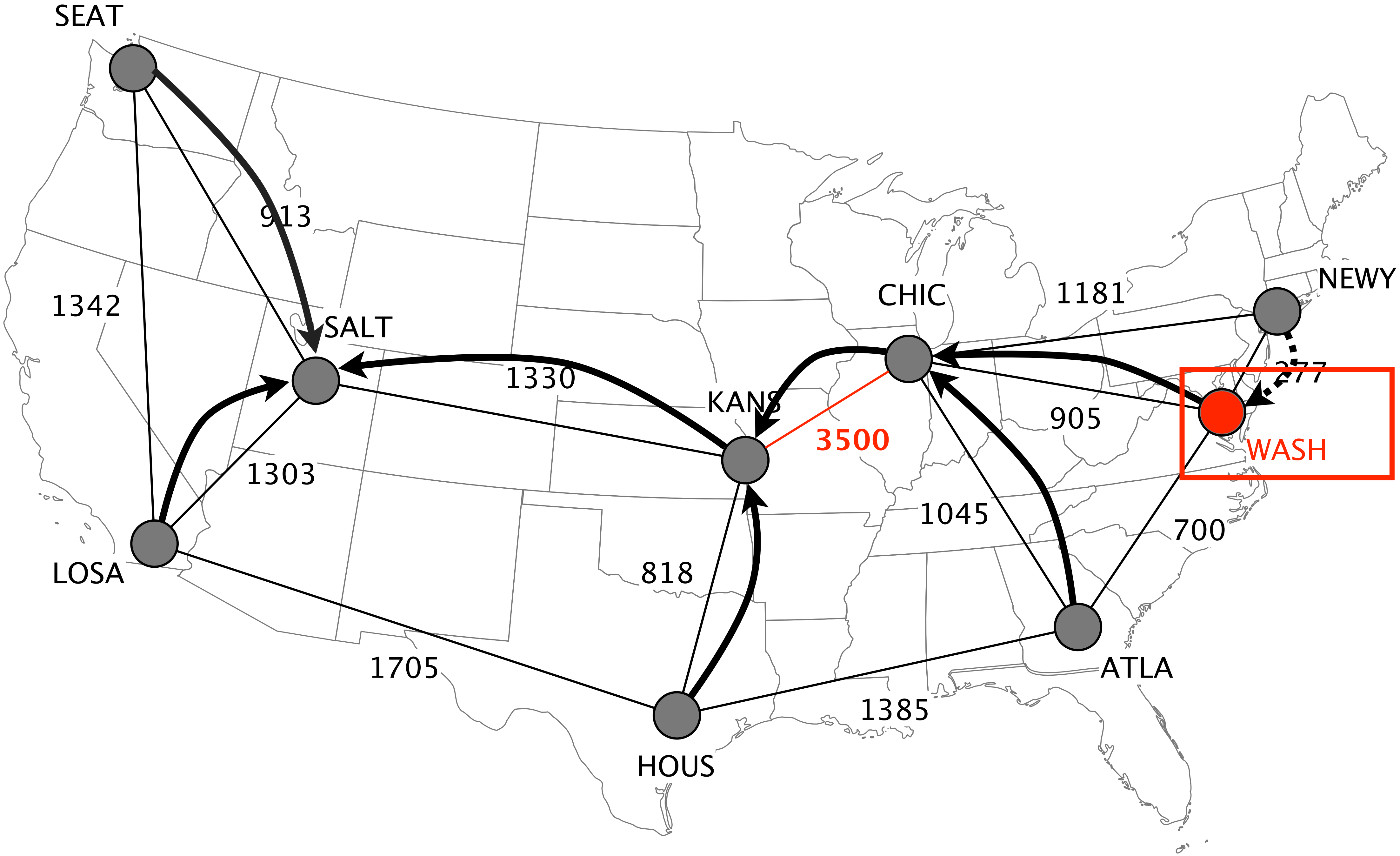
Migrated [NEWY]

To migrate [WASH, CHIC, ATLA, ...]



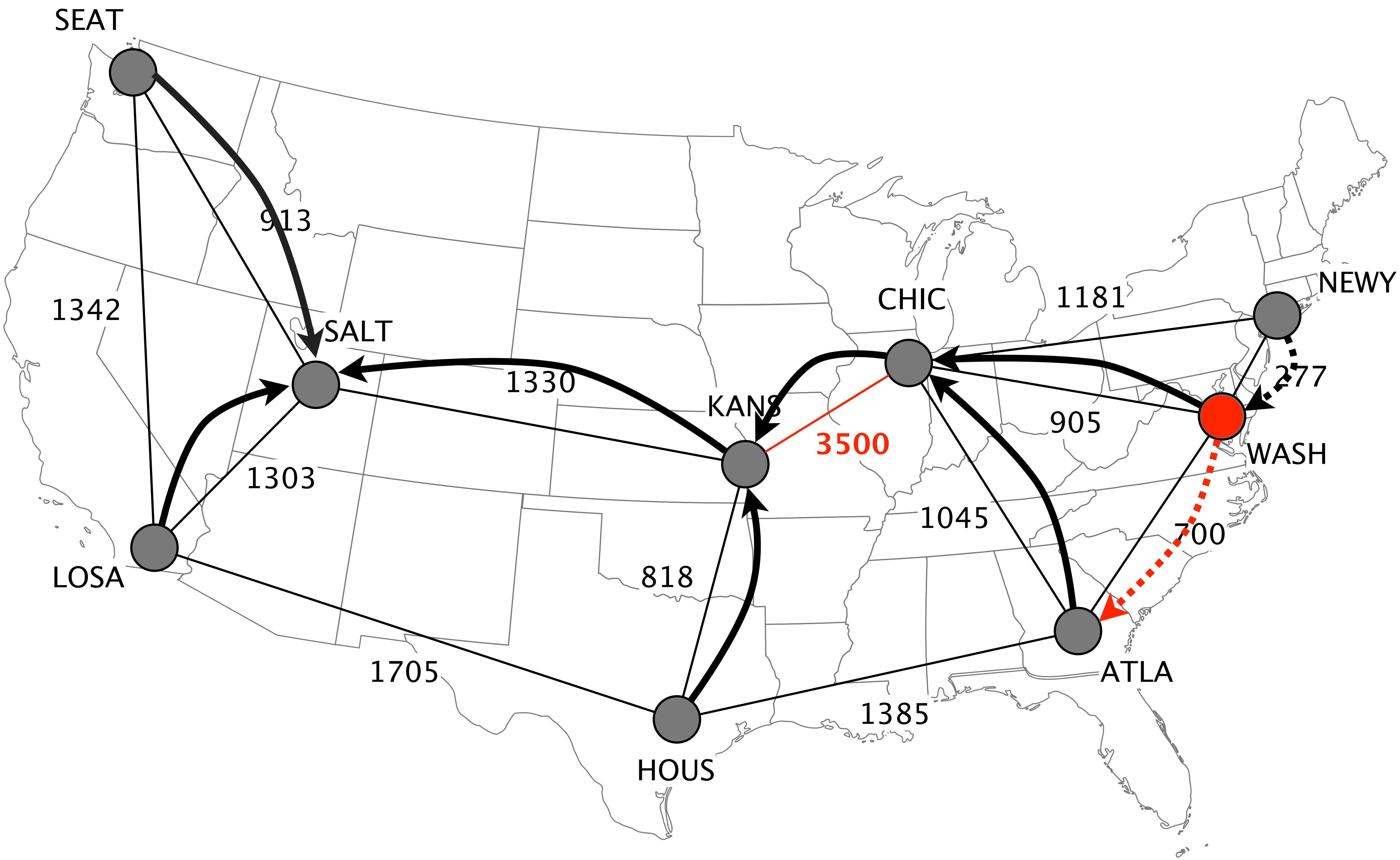
Migrated [NEWY]

To migrate [WASH, CHIC, ATLA, ...]

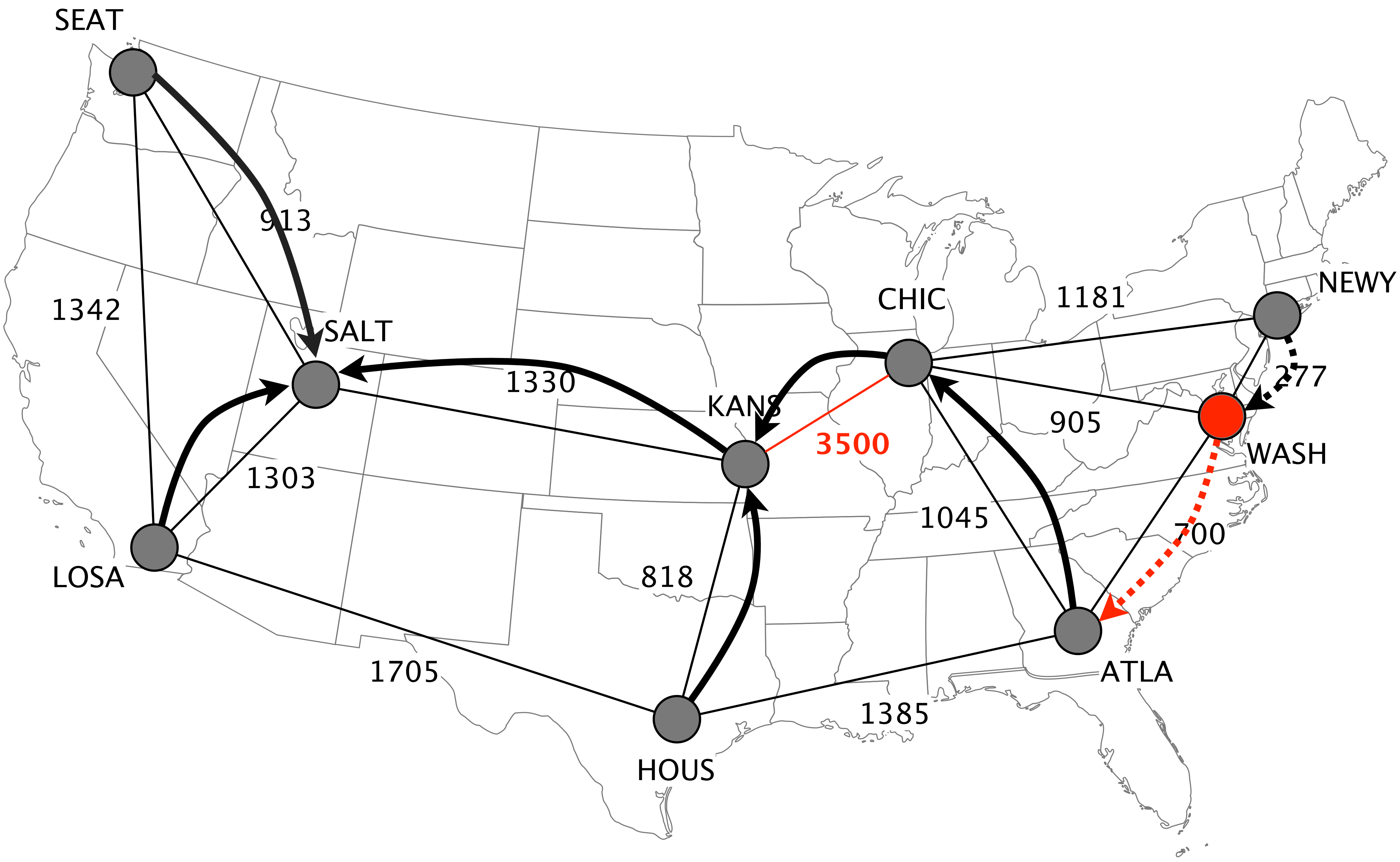


Migrated [NEWY]

To migrate [WASH, CHIC, ATLA, ...]

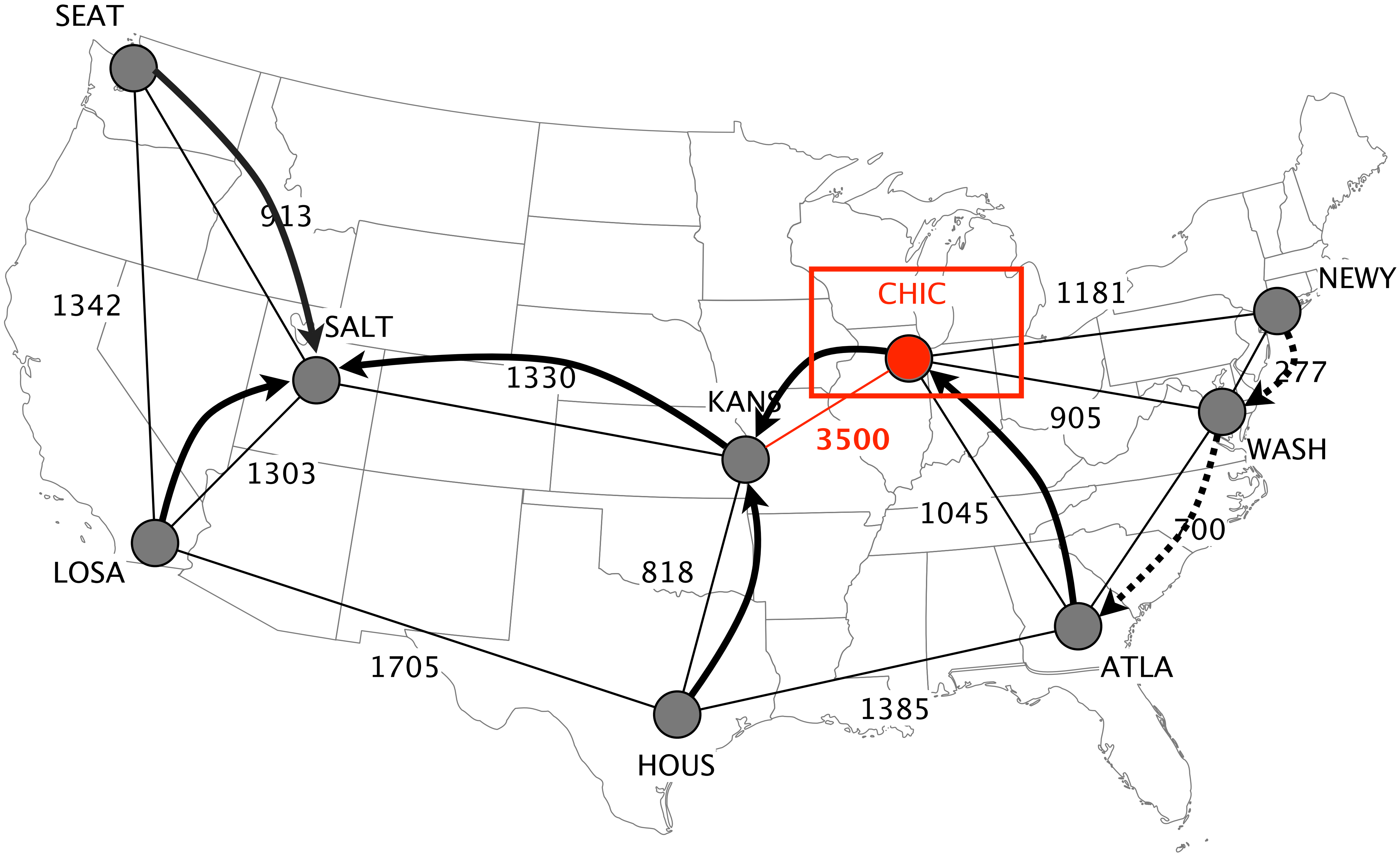


Migrated	[NEWY, WASH]
To migrate	[CHIC, ATLA, ...]

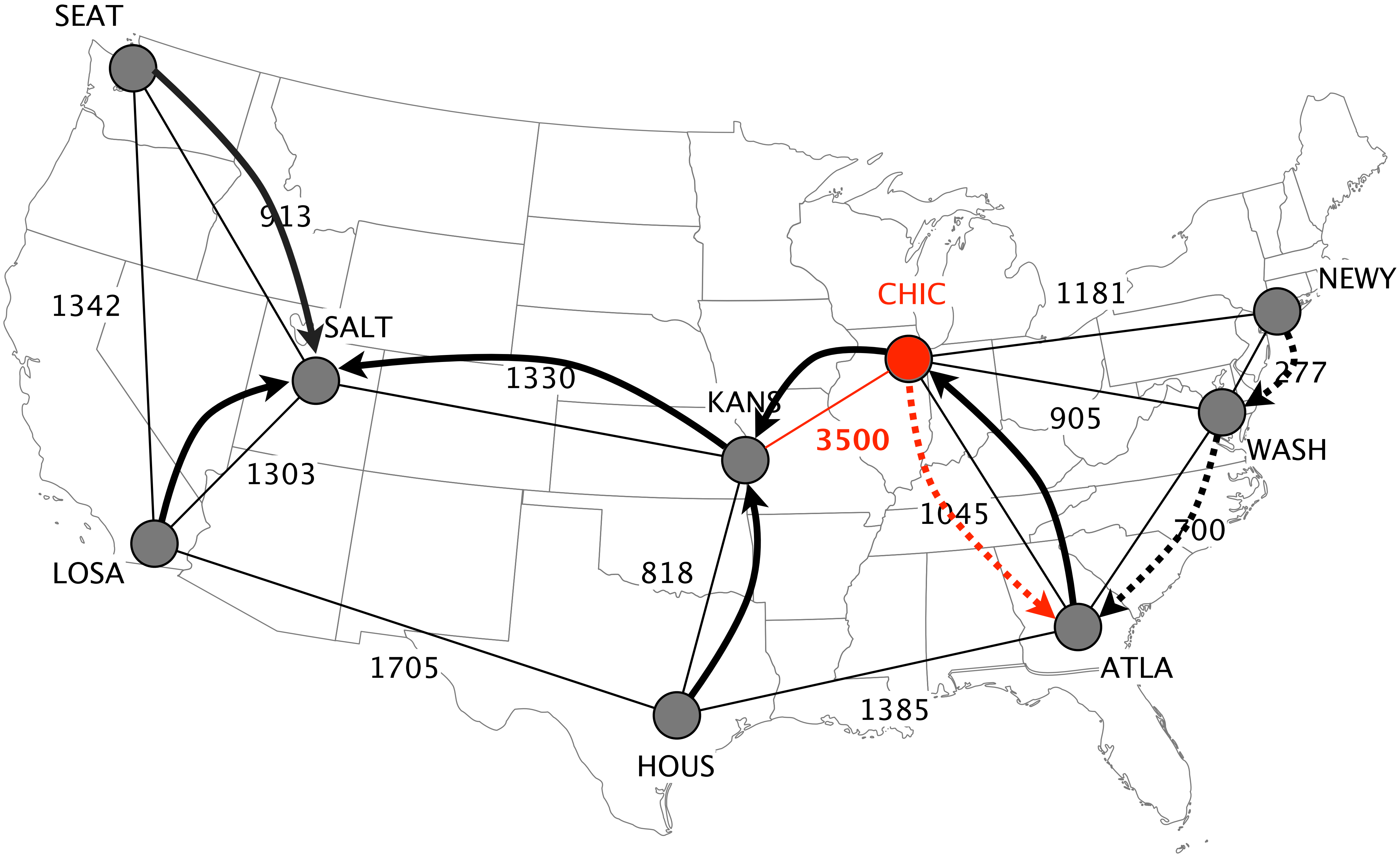


Migrated [NEWY, WASH]

To migrate [CHIC, ATLA, ...]

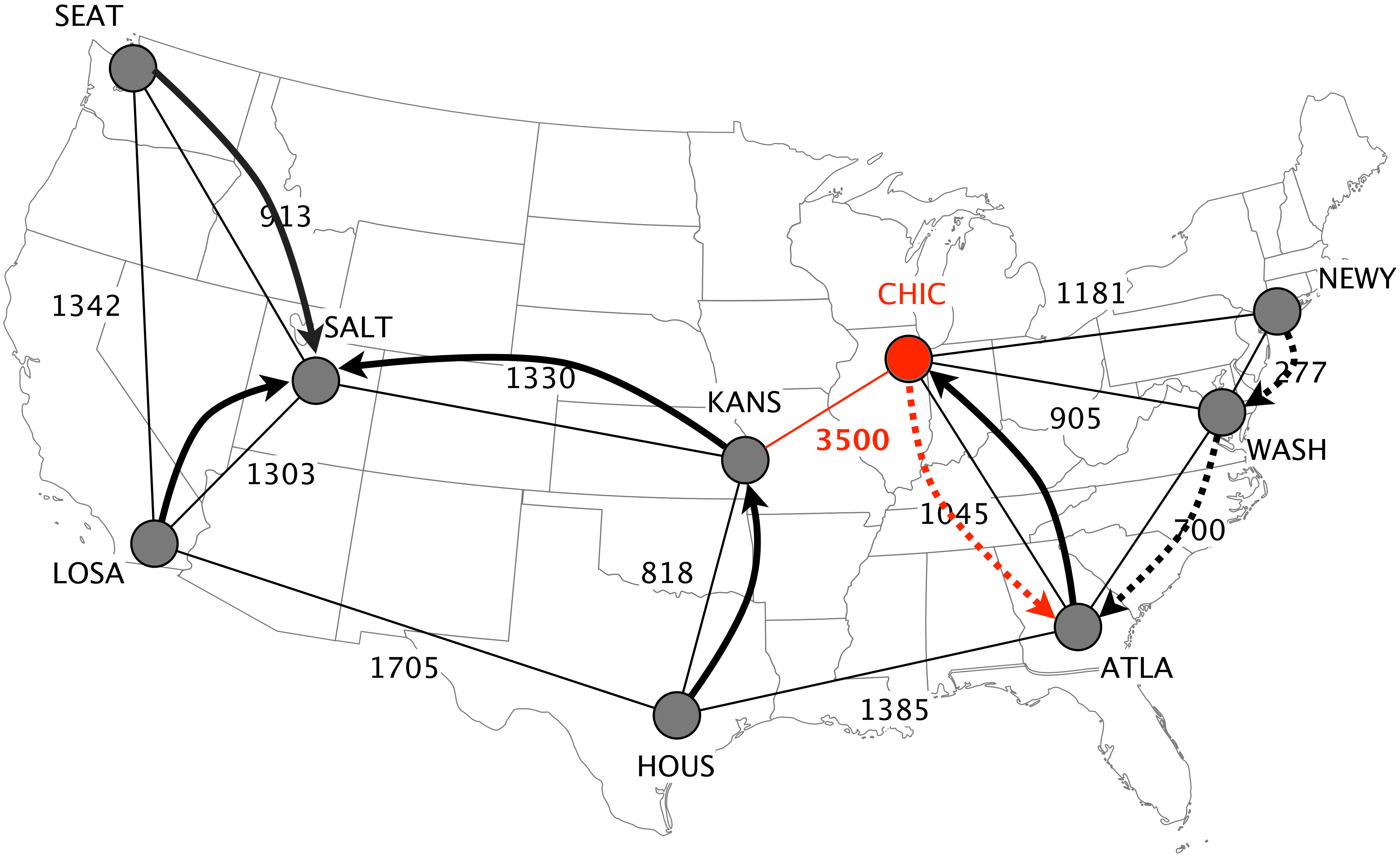


Migrated	[NEWY, WASH]
To migrate	[CHIC, ATLA, ...]



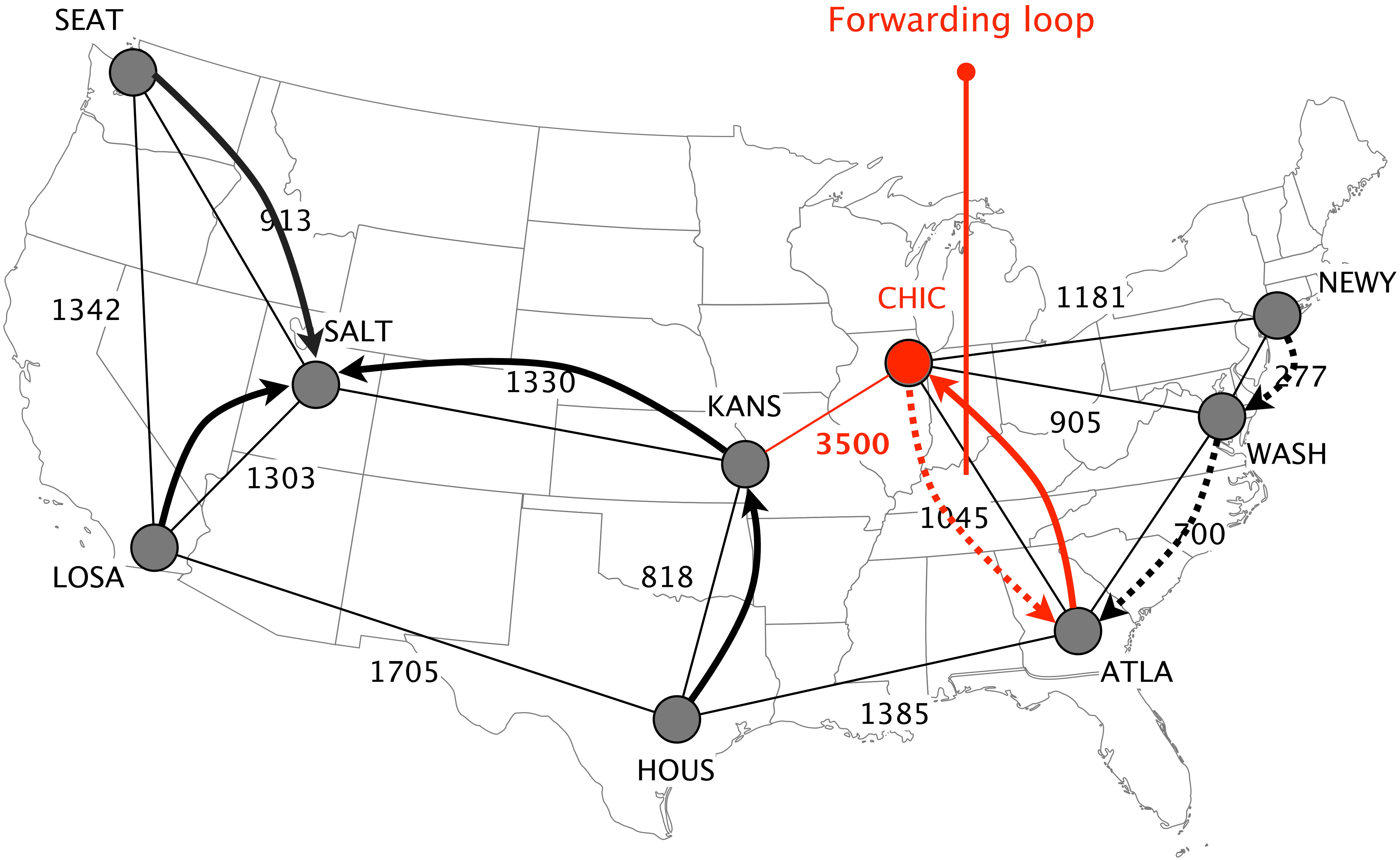
Migrated [NEWY, WASH, CHIC]

To migrate [ATLA, ...]



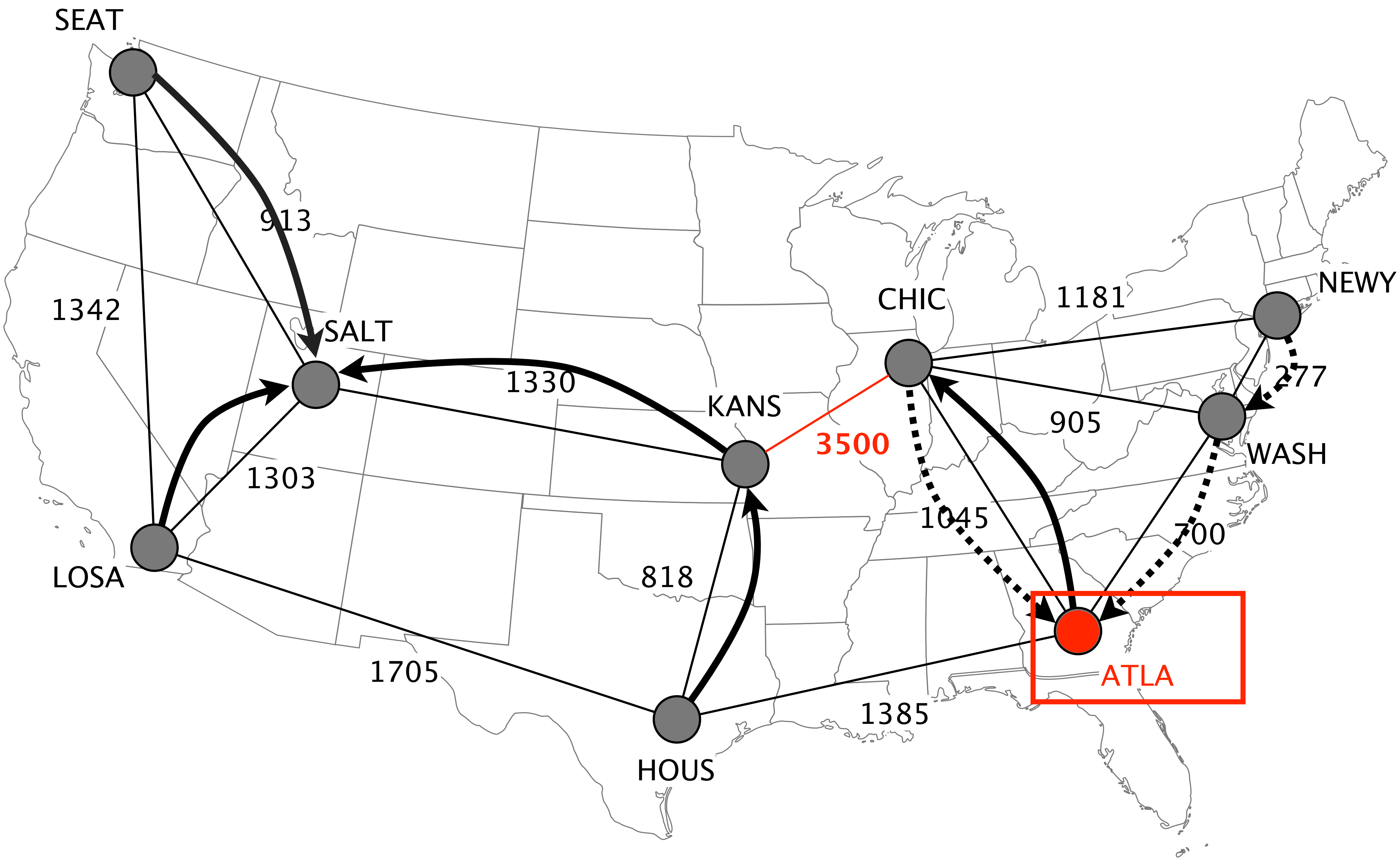
Migrated [NEWY, WASH, CHIC]

To migrate [ATLA, ...]



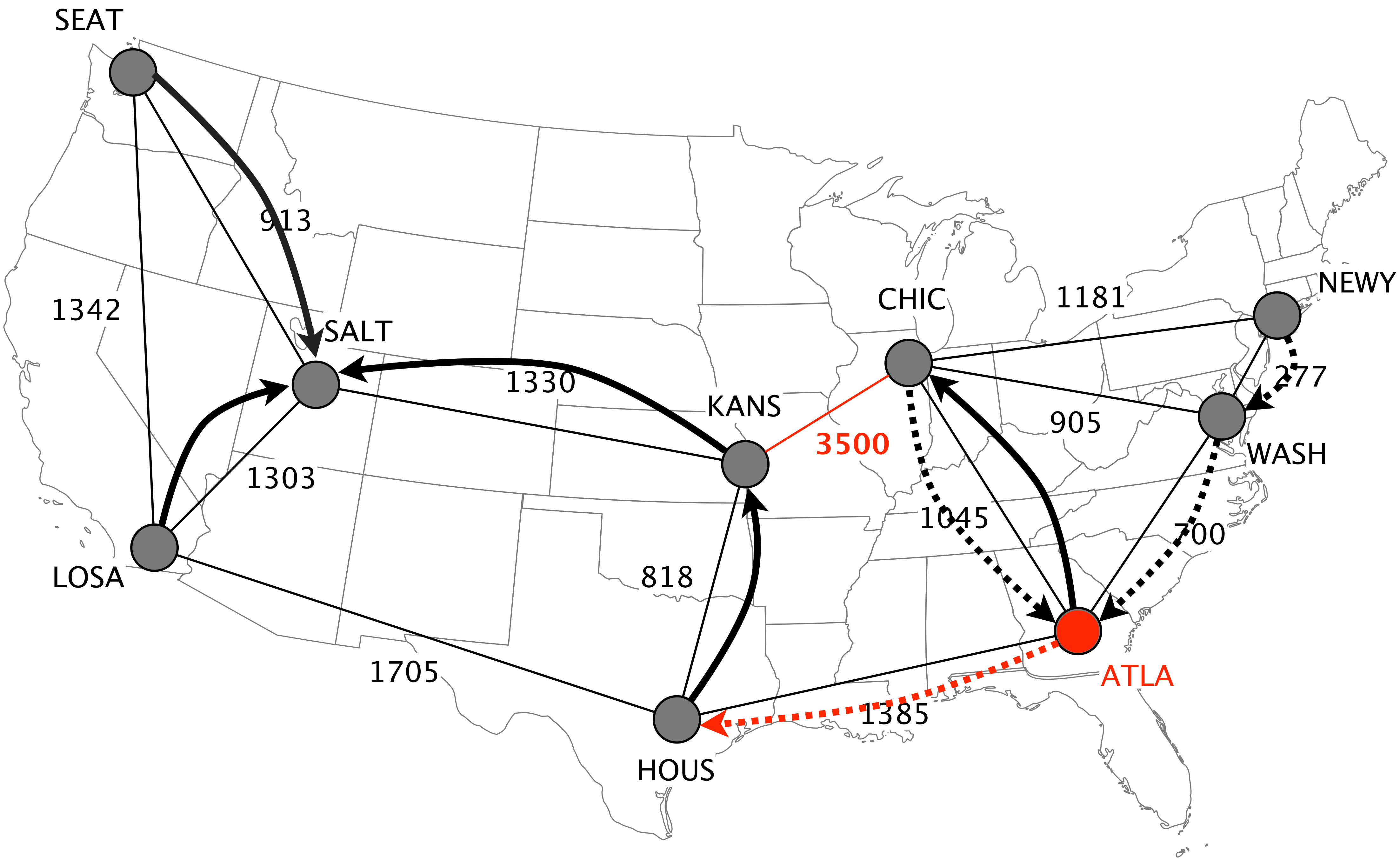
Migrated [NEWY, WASH, CHIC]

To migrate [ATLA, ...]



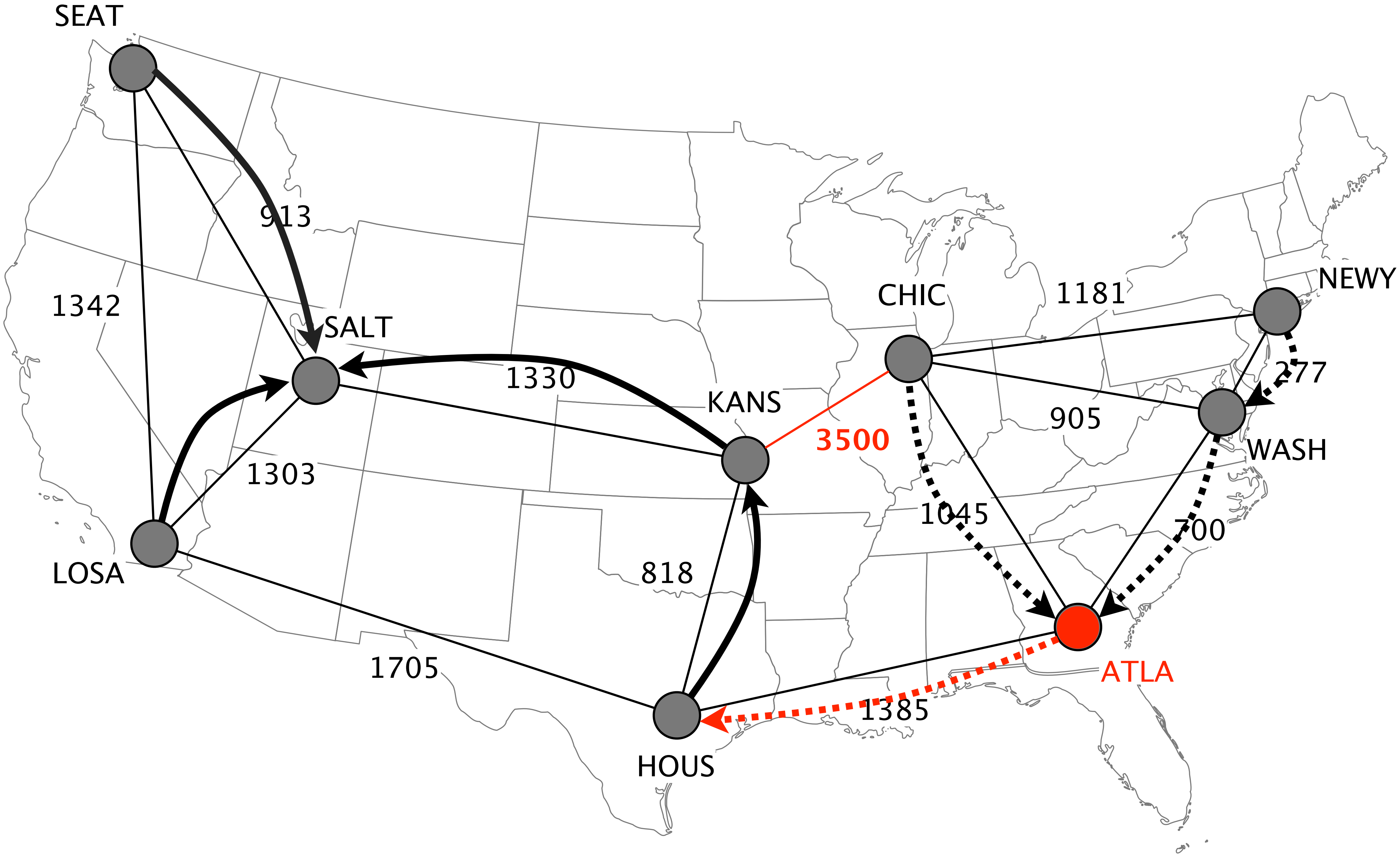
Migrated [NEWY, WASH, CHIC]

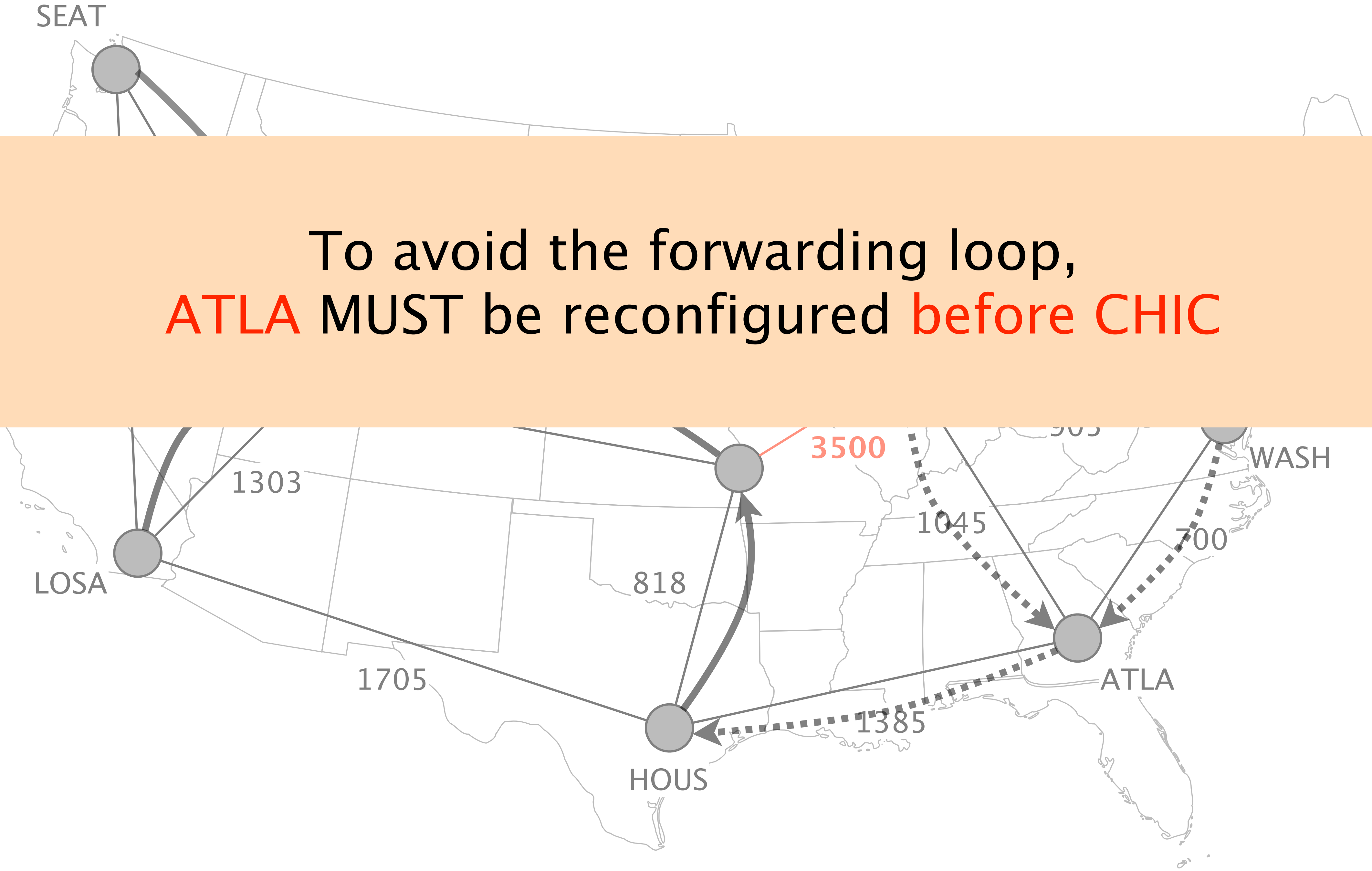
To migrate [ATLA, ...]



Migrated [NEWY, WASH, CHIC, ATLA]

To migrate []

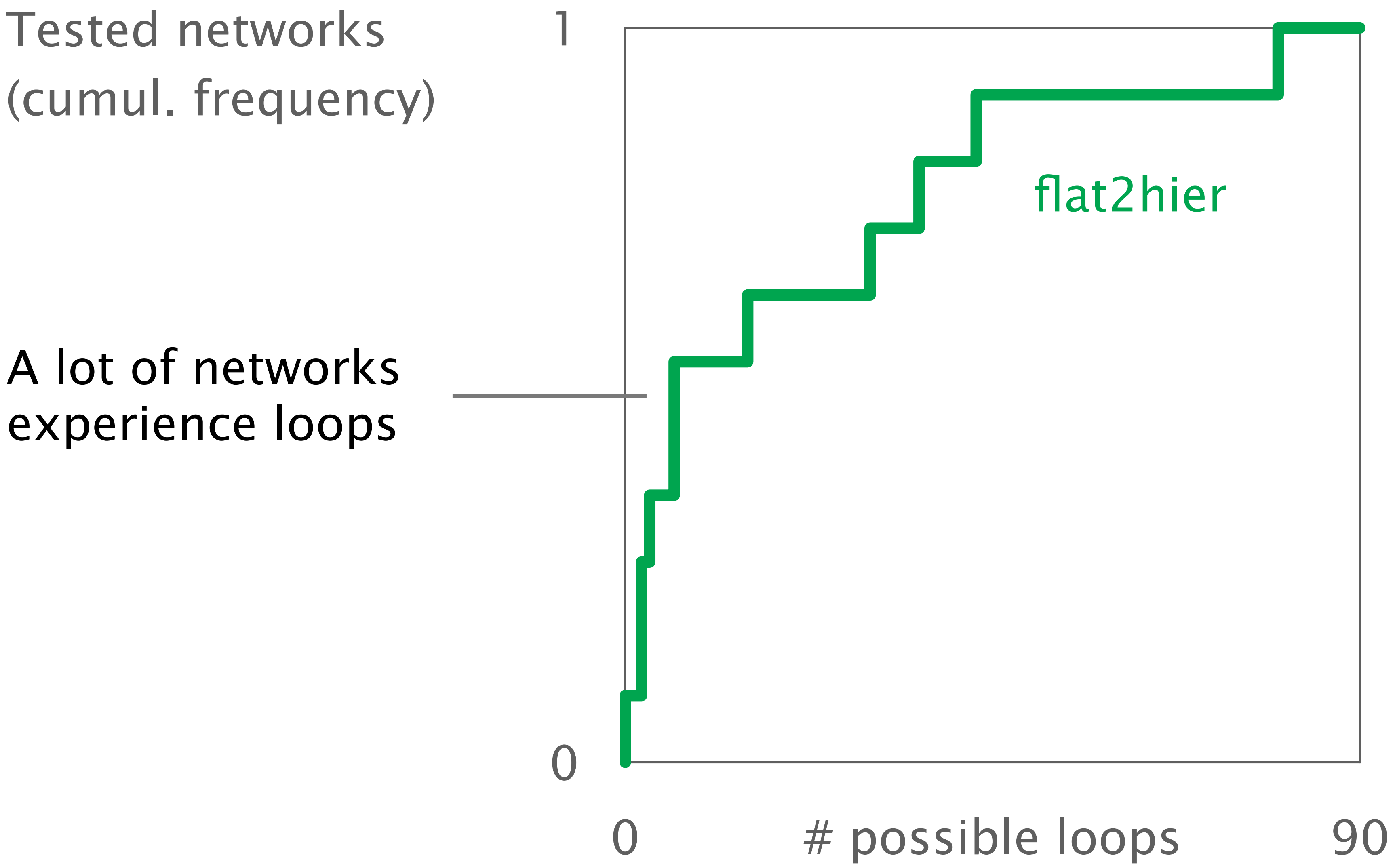




To avoid the forwarding loop,
ATLA MUST be reconfigured before CHIC

But ... are forwarding loops such an issue?

Numerous forwarding loops can appear in LS to LS reconfigurations



A lot of networks experience loops

Up to 80 *reconfiguration loops* can arise during an IGP migration

Find an ordering in which to activate the final IGP
without causing any forwarding anomalies

Find an ordering in which to activate the final IGP without causing any forwarding anomalies

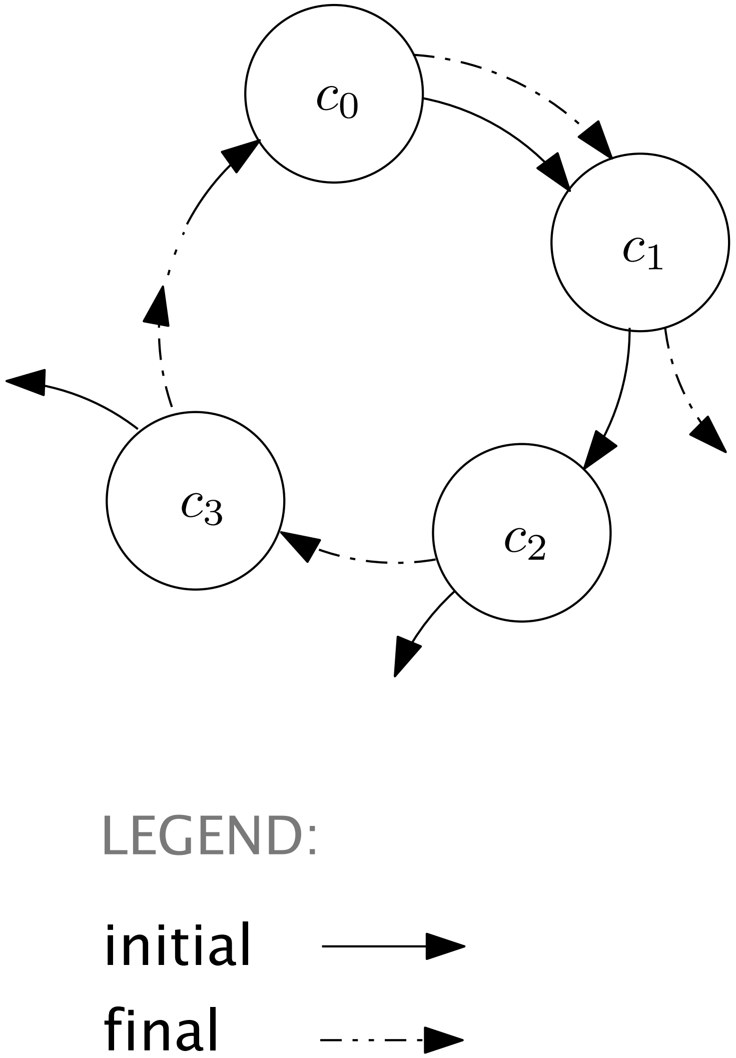
Is it easy to compute?

Find an ordering in which to activate the final IGP
without causing any forwarding anomalies

Is it easy to compute?

Does it always exist?

Deciding if an ordering exists is computationally hard (NP-complete)



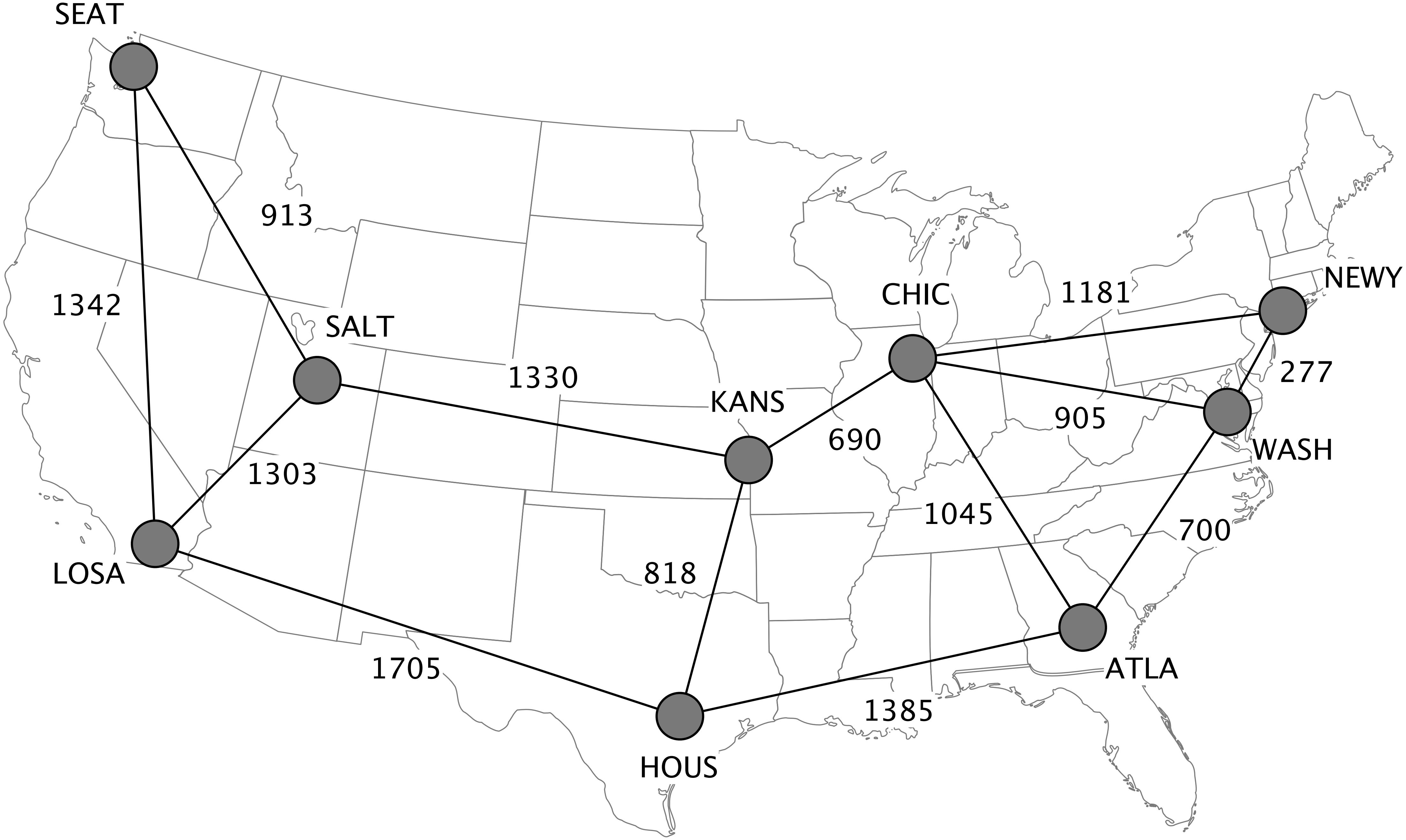
The Enumeration Algorithm [correct & complete]

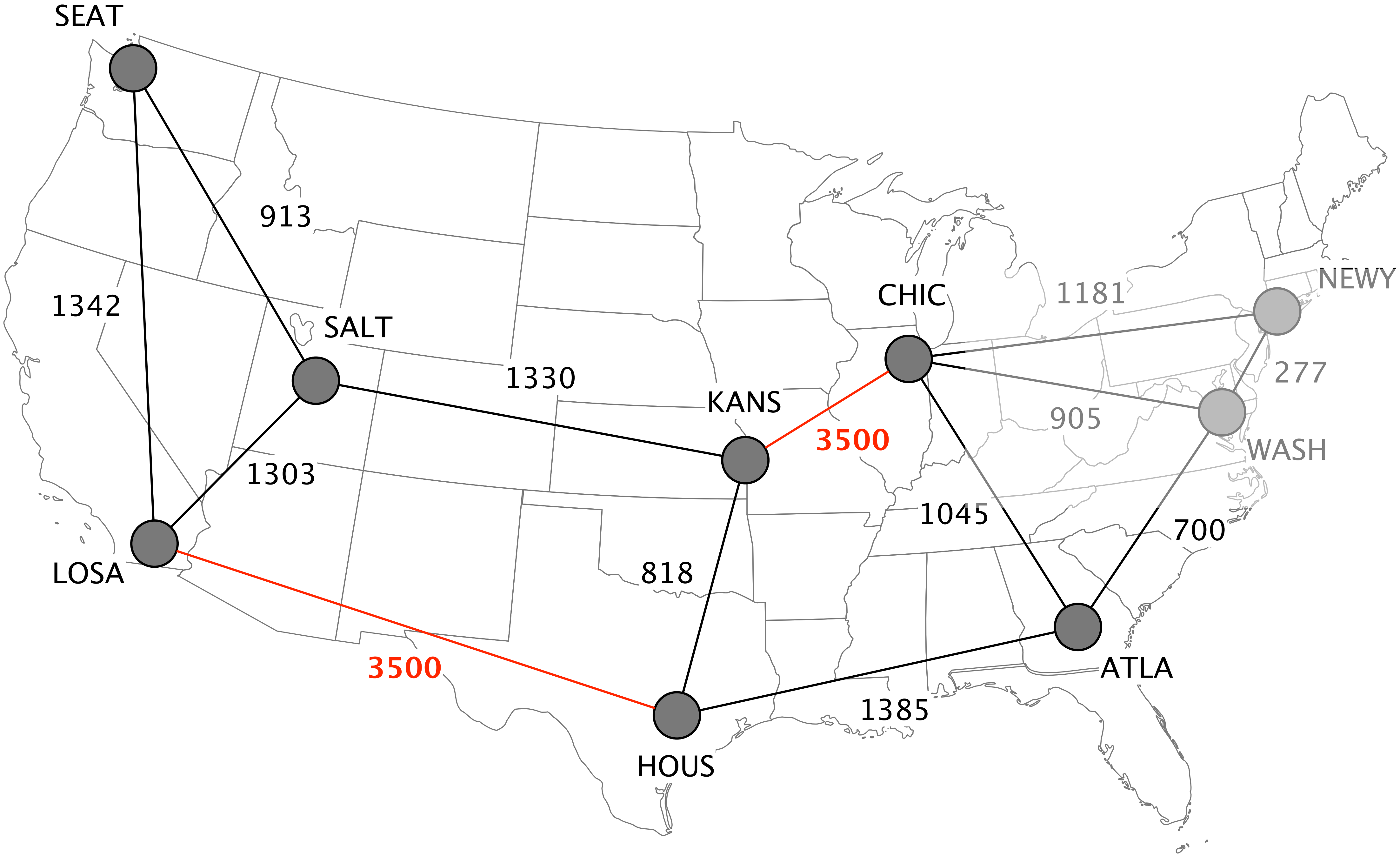
1. Merge the initial and the final forwarding paths
2. For each migration loop in the merged graph,

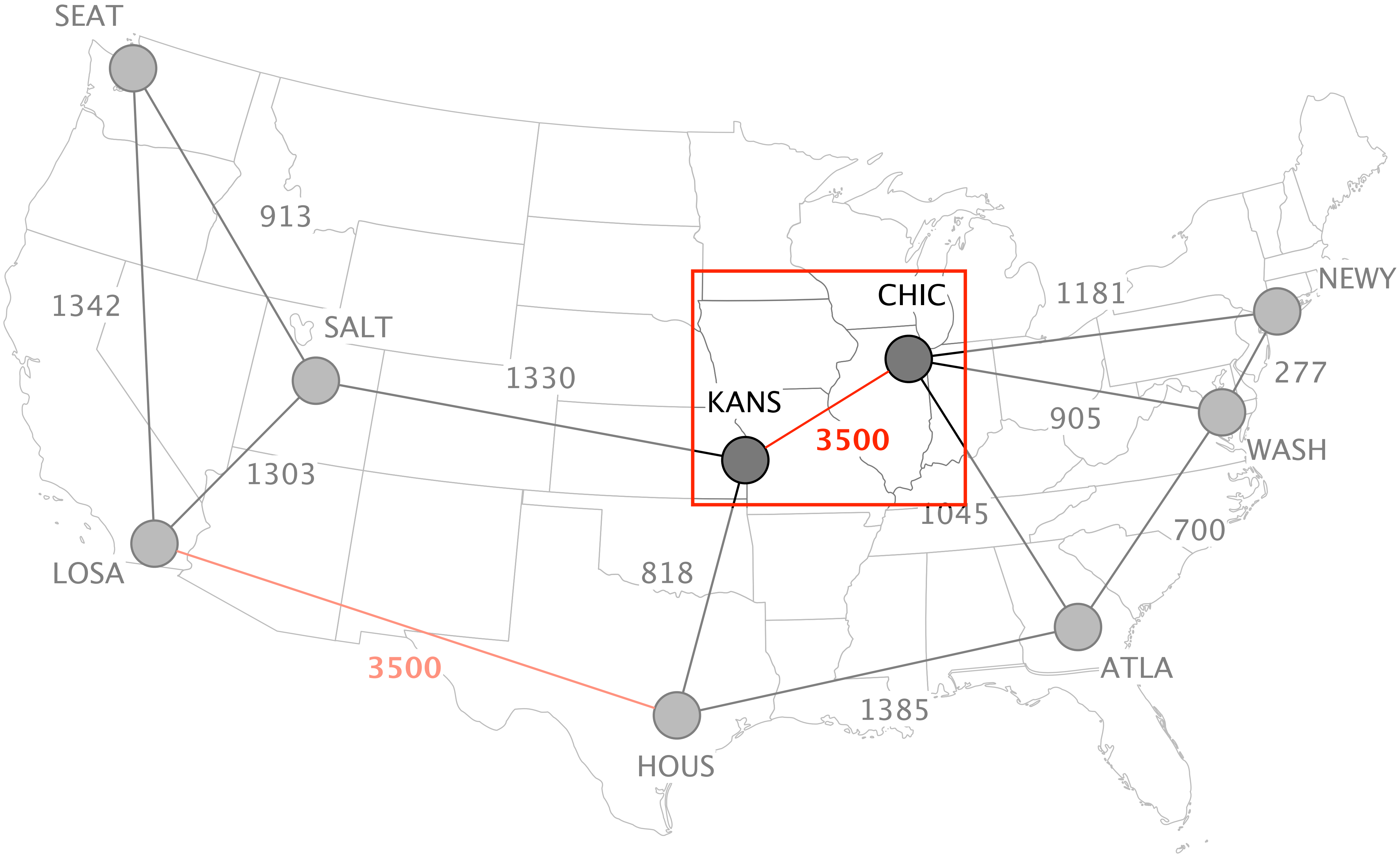
    Output ordering constraints such that
    at least one router in the initial state
    is migrated before at least one in the final

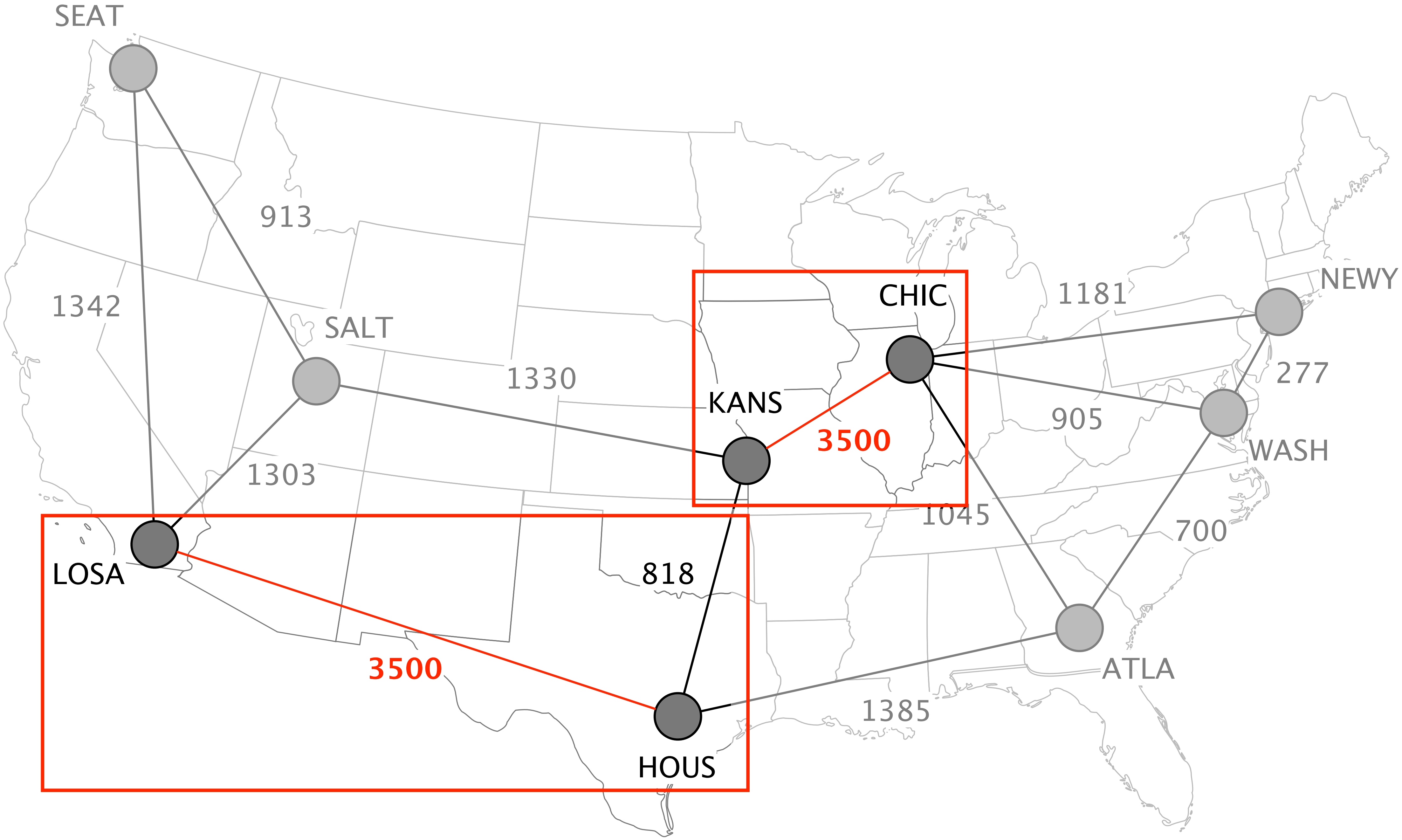
3. Solve the system by using Linear Programming

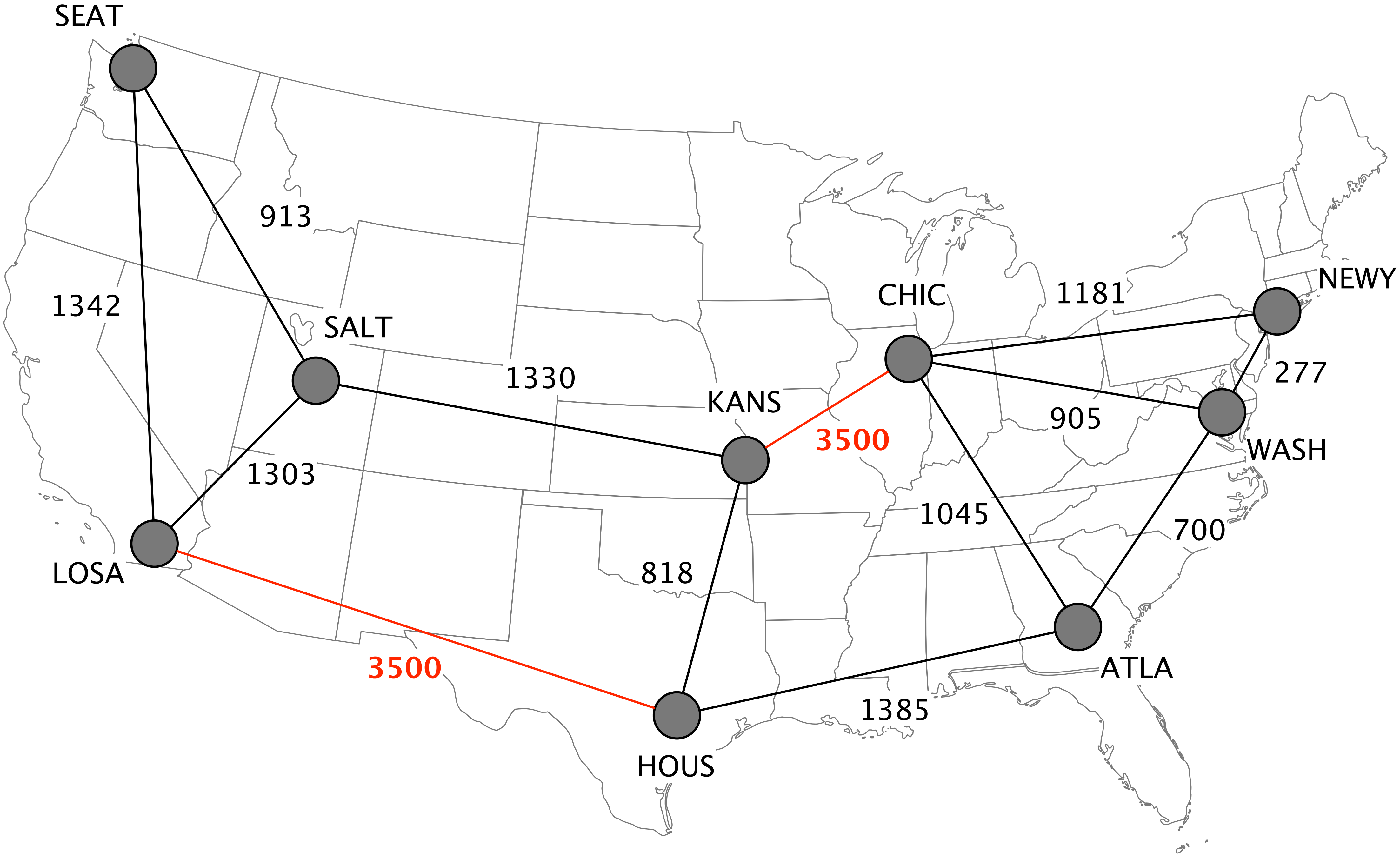
Due to contradictory constraints, an ordering does not always exist



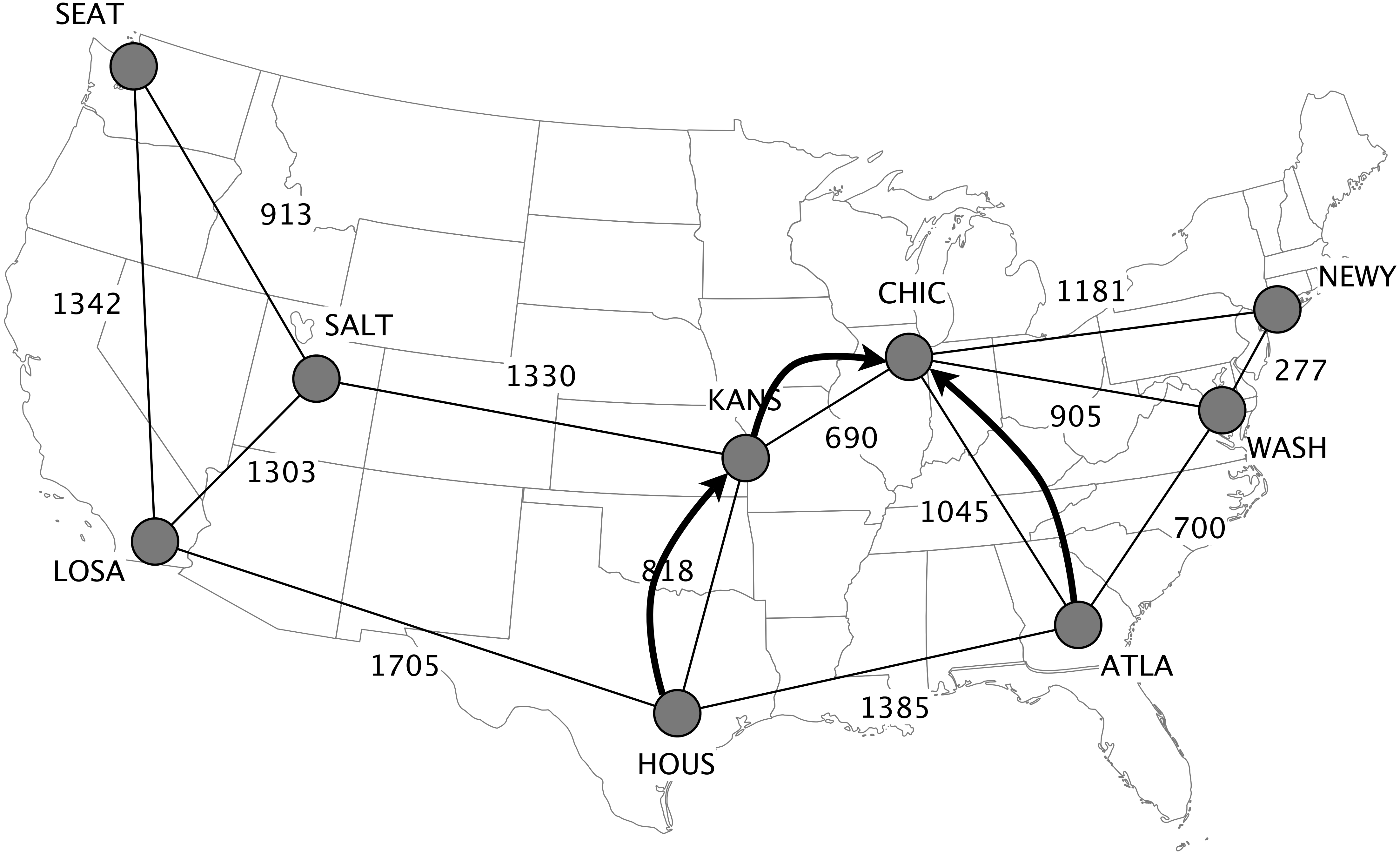

SEAT
913
1342
SALT
1330
1303
KANS
3500
CHIC
1181
NEWY
277
905
WASH
LOSA
1045
700
818
3500
ATLA
1385
HOUS


SEAT
913
1342
SALT
1330
KANS
CHIC
1181
NEWY
277
905
WASH
3500
1303
LOSA
1045
700
818
3500
ATLA
1385
HOUS

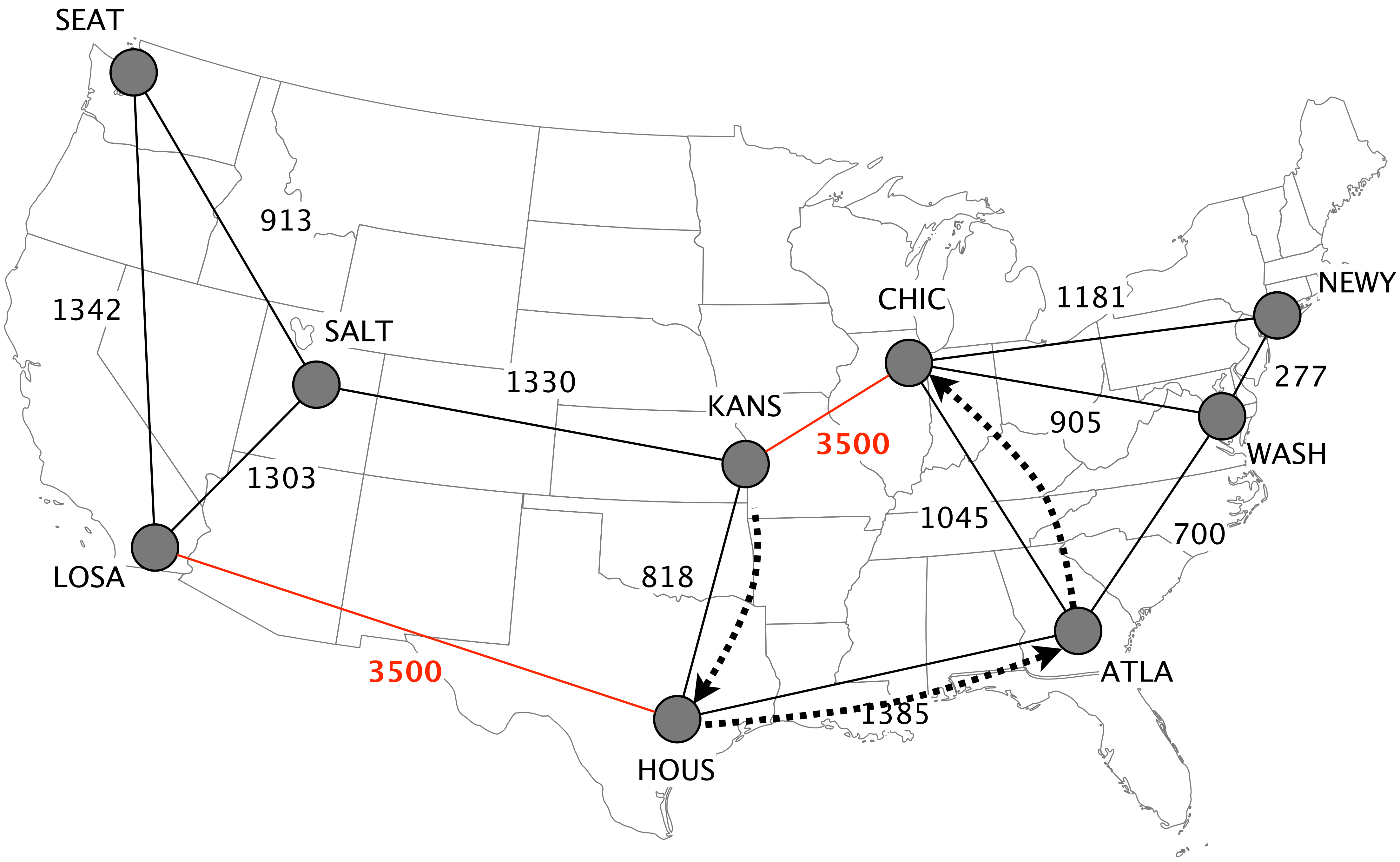

SEAT
913
1342
SALT
1330
CHIC
1181
NEWY
KANS
277
3500
905
WASH
1303
1045
700
LOSA
818
3500
ATLA
1385
HOUS

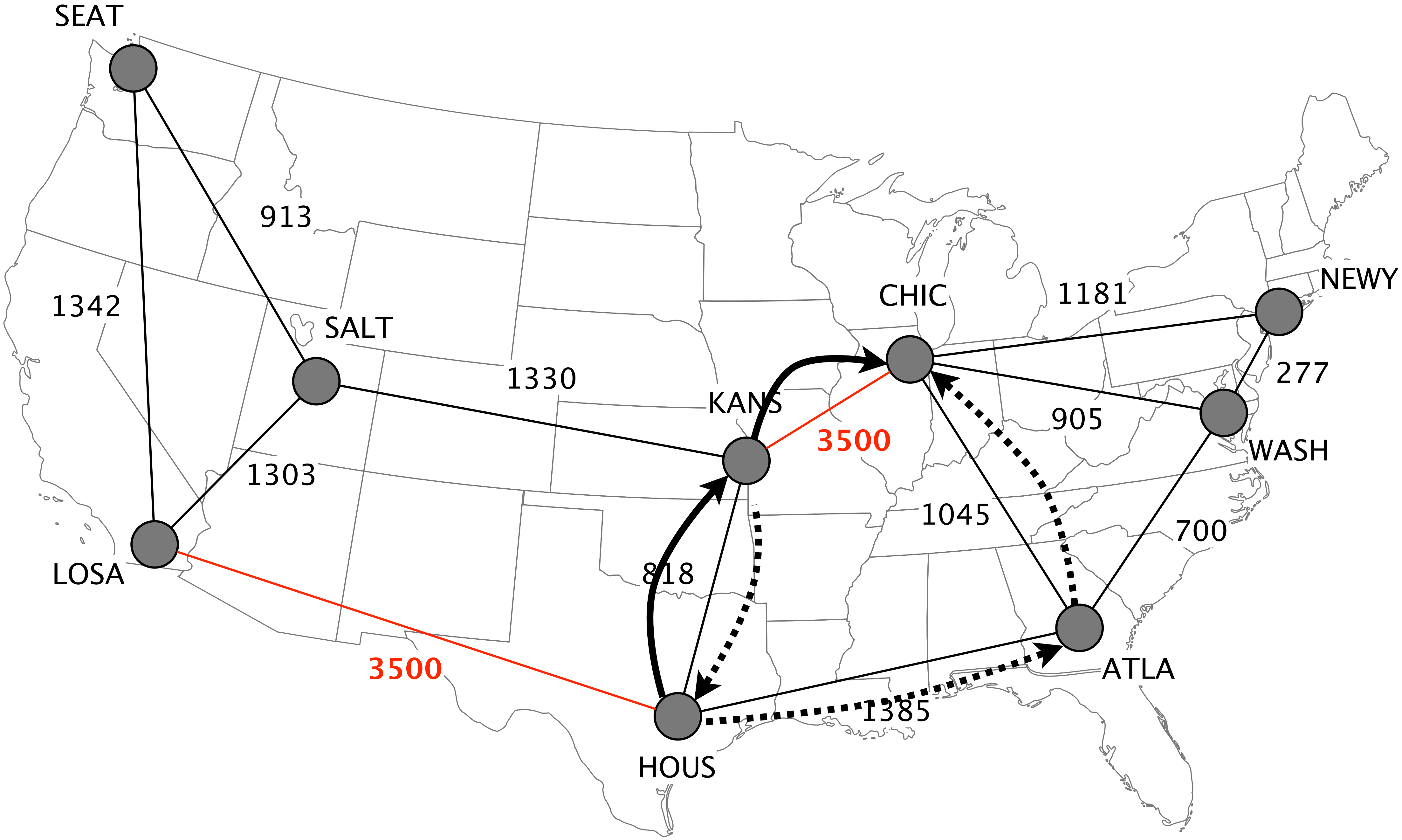

SEAT
913
1342
SALT
1330
KANS
CHIC
1181
NEWY
277
905
WASH
3500
1303
1045
700
LOSA
818
ATLA
3500
1385
HOUS

Initial forwarding paths towards CHIC

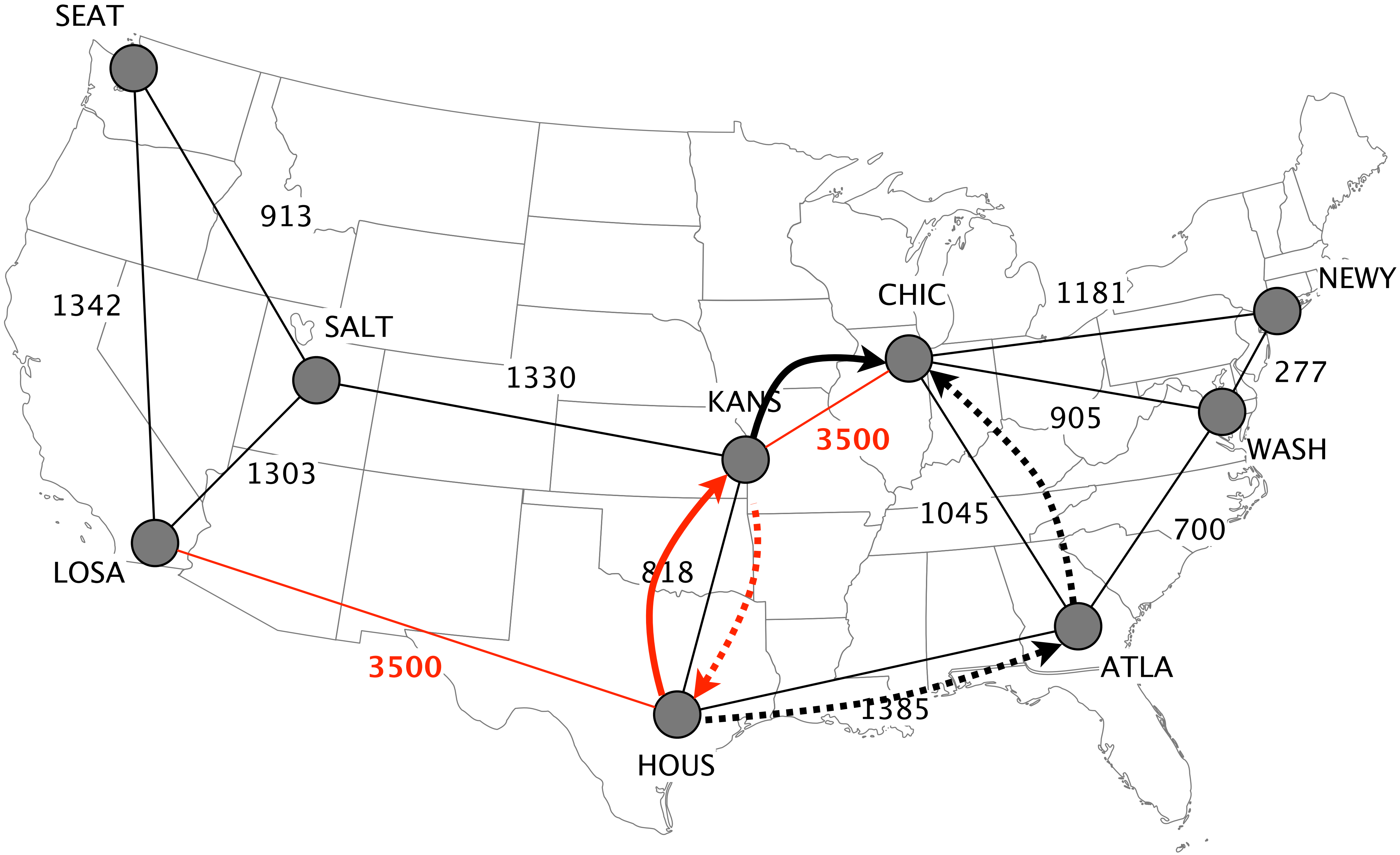


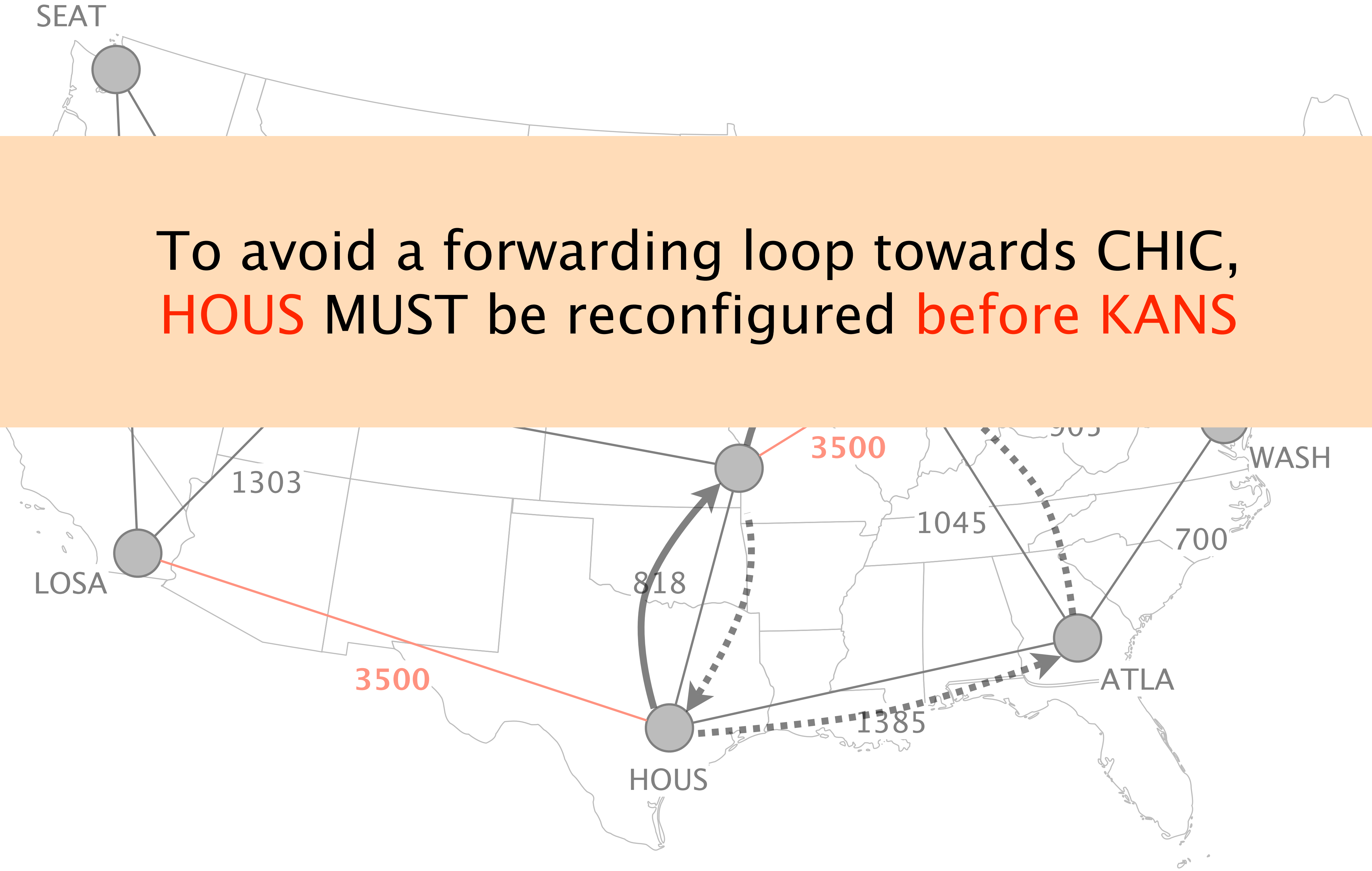
Final forwarding paths towards CHIC




SEAT
913
1342
SALT
1330
1303
LOSA
3500
KANS
818
HOUS
1385
CHIC
3500
1181
NEWY
277
905
WASH
1045
700
ATLA

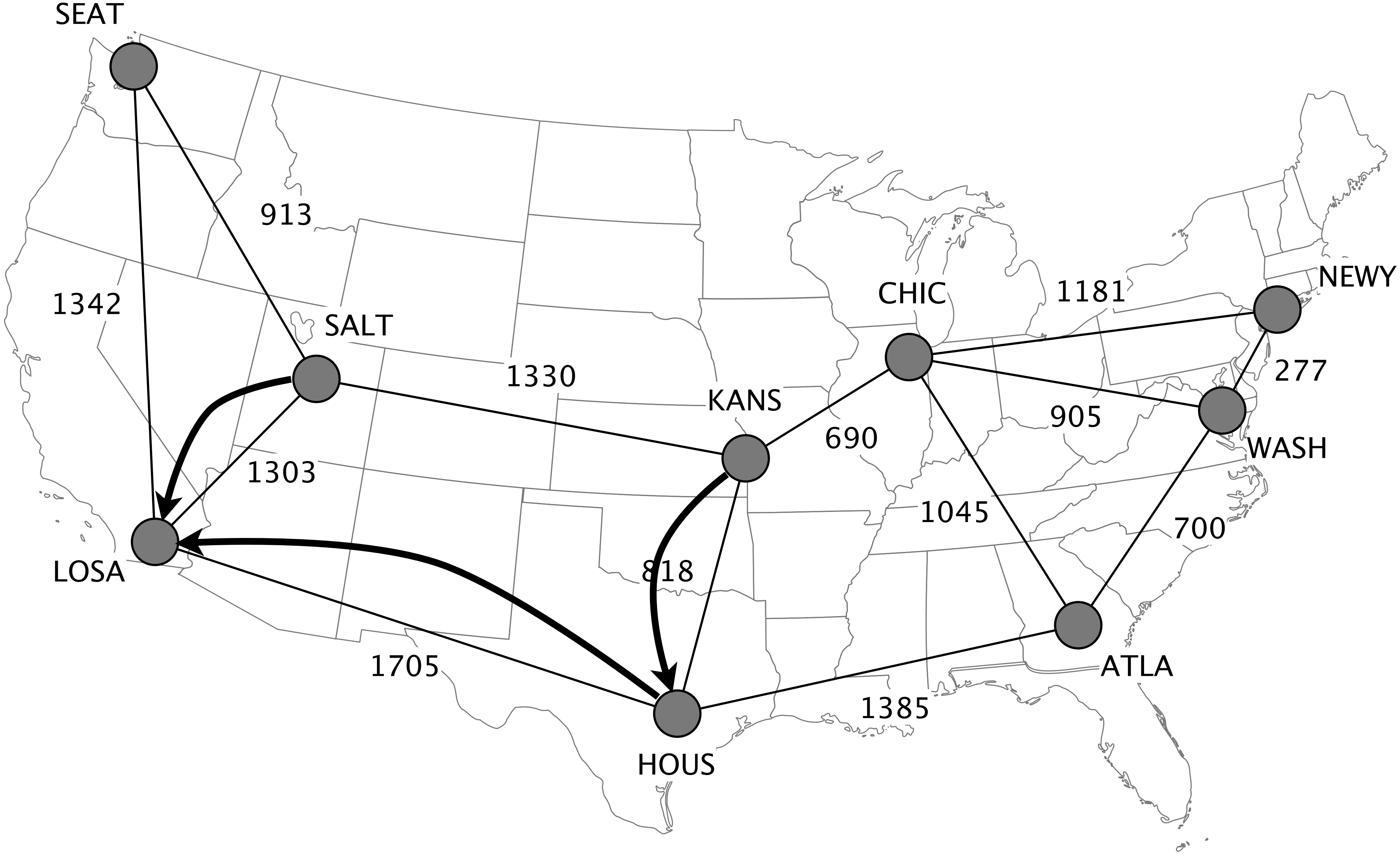
Merged forwarding paths towards CHIC



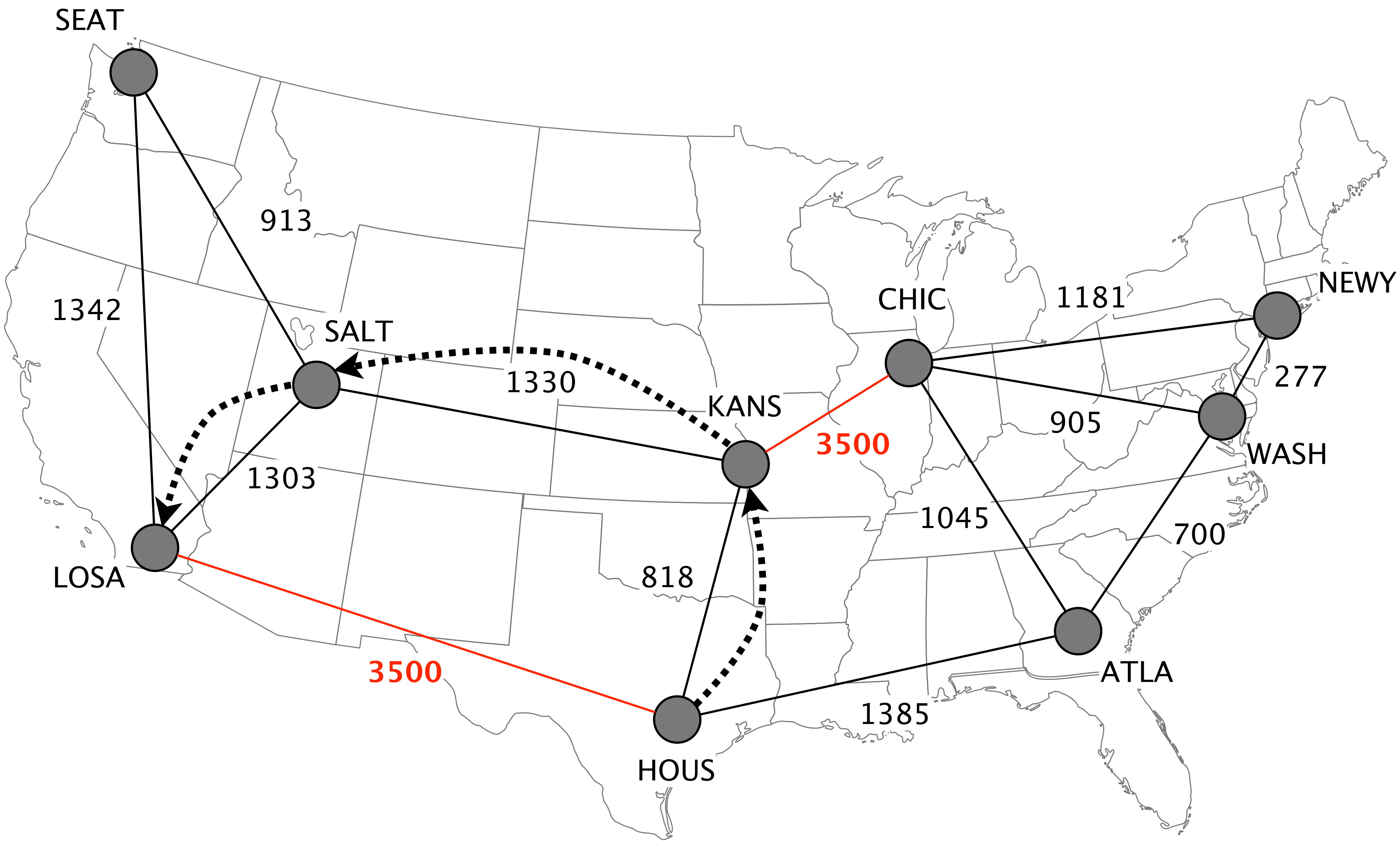


To avoid a forwarding loop towards CHIC,
HOUS MUST be reconfigured before KANS

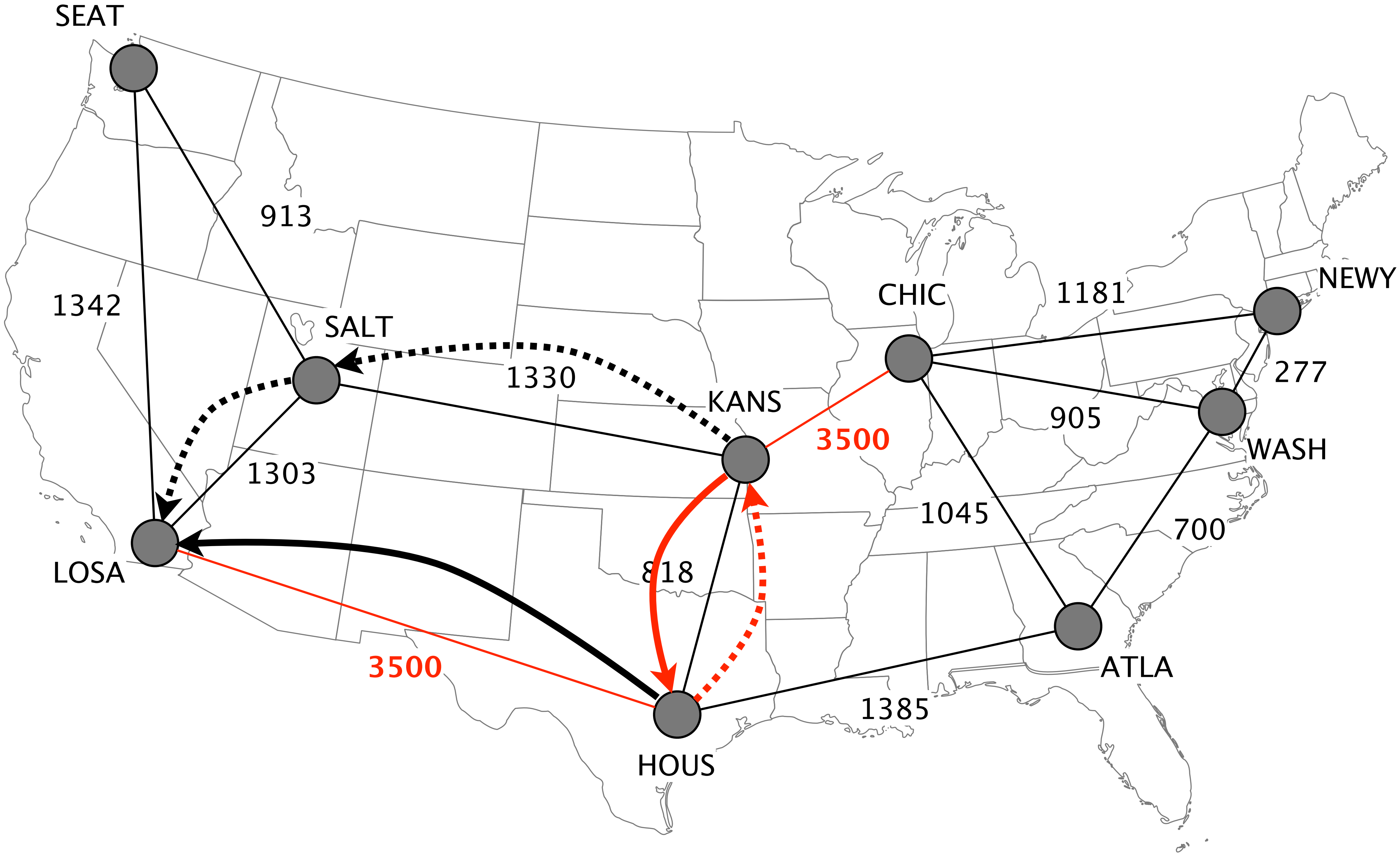
Initial forwarding paths towards LOSA



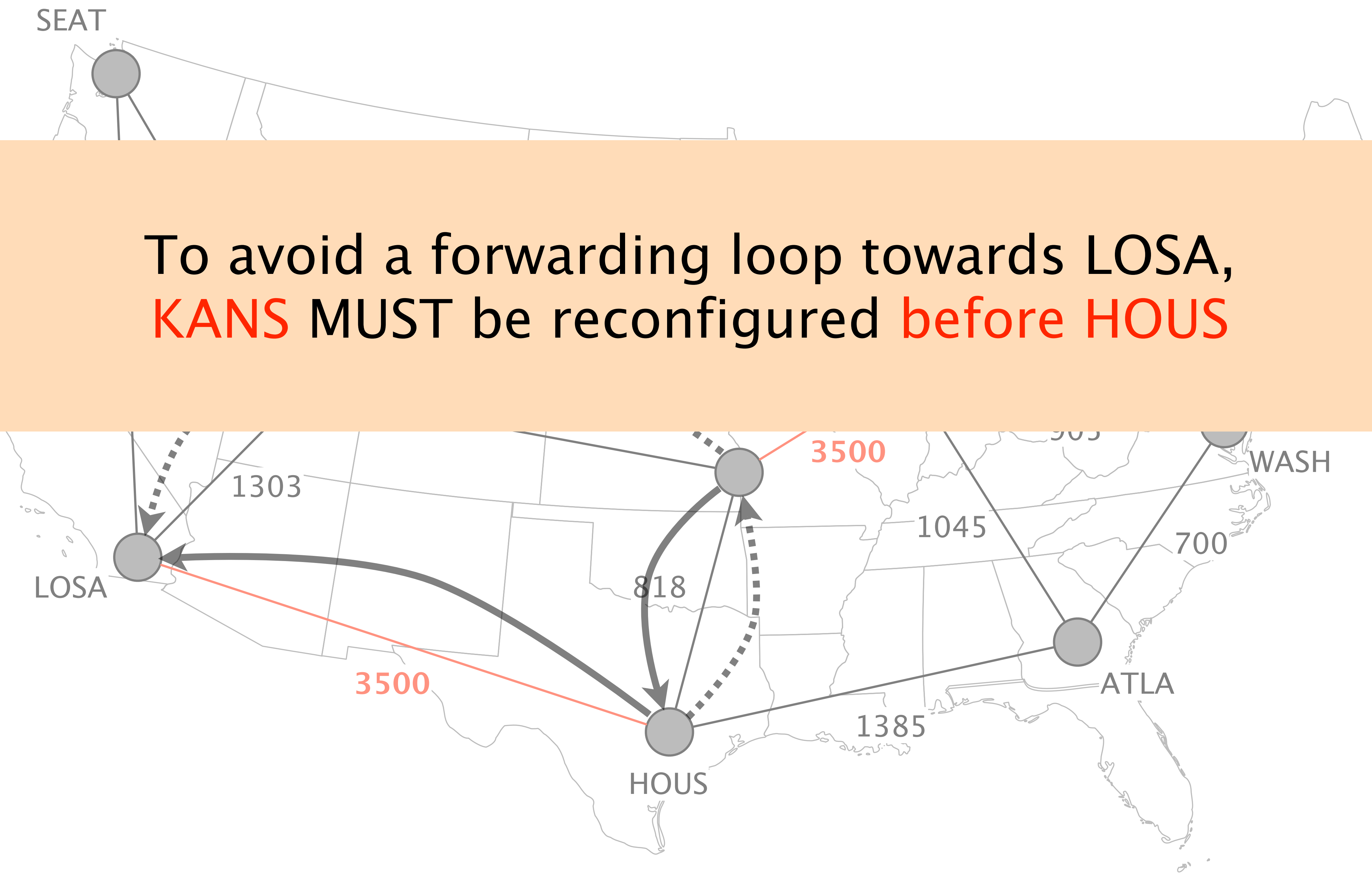
Final forwarding paths towards LOSA



Merged forwarding paths towards LOSA



To avoid a forwarding loop towards LOSA,
KANS MUST be reconfigured before HOUS



Due to contradictory constraints, an ordering does not always exist

One of these constraints will not be met:

HOUS < KANS

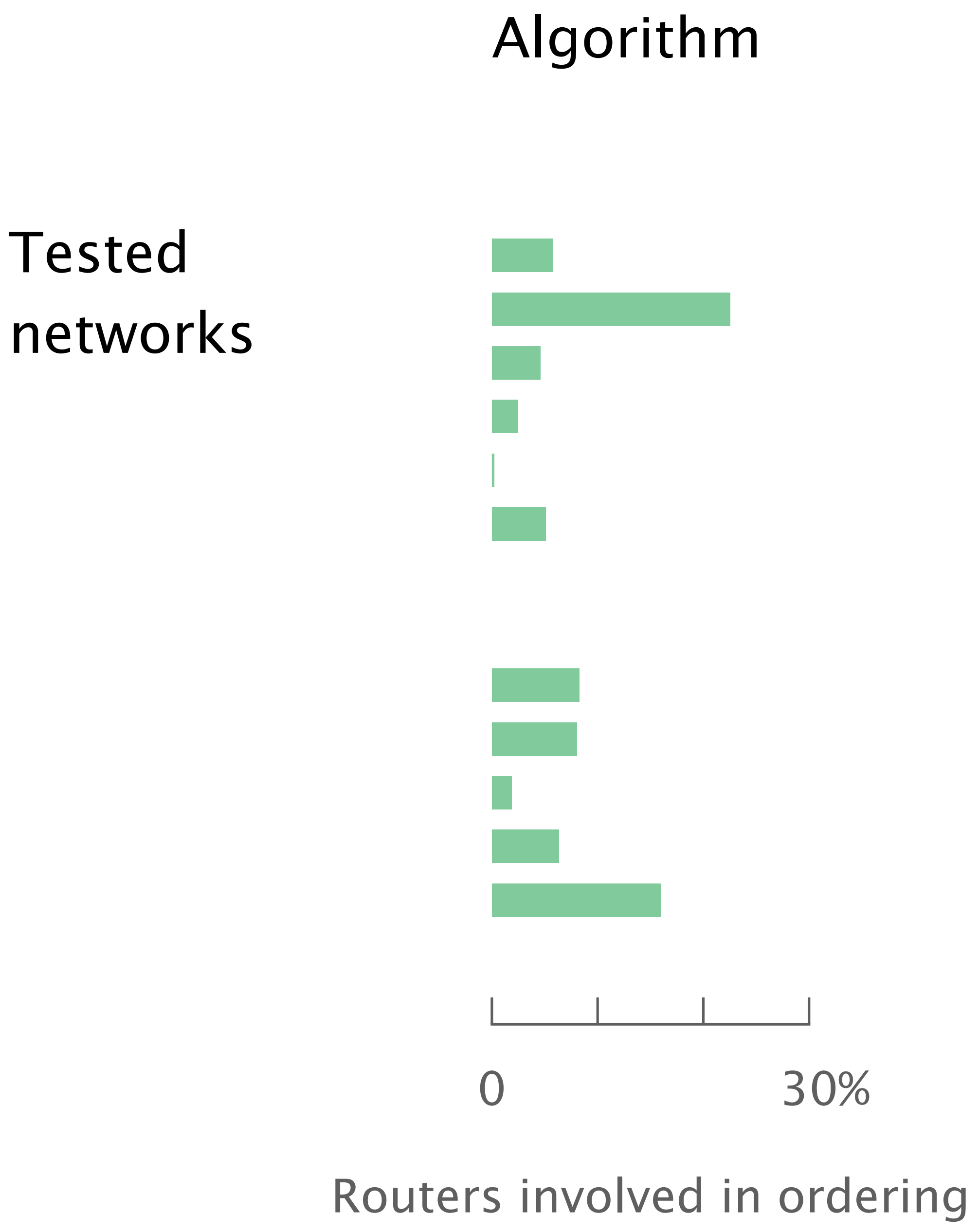
To avoid a forwarding loop towards CHIC,
HOUS MUST be reconfigured before KANS

KANS < HOUS

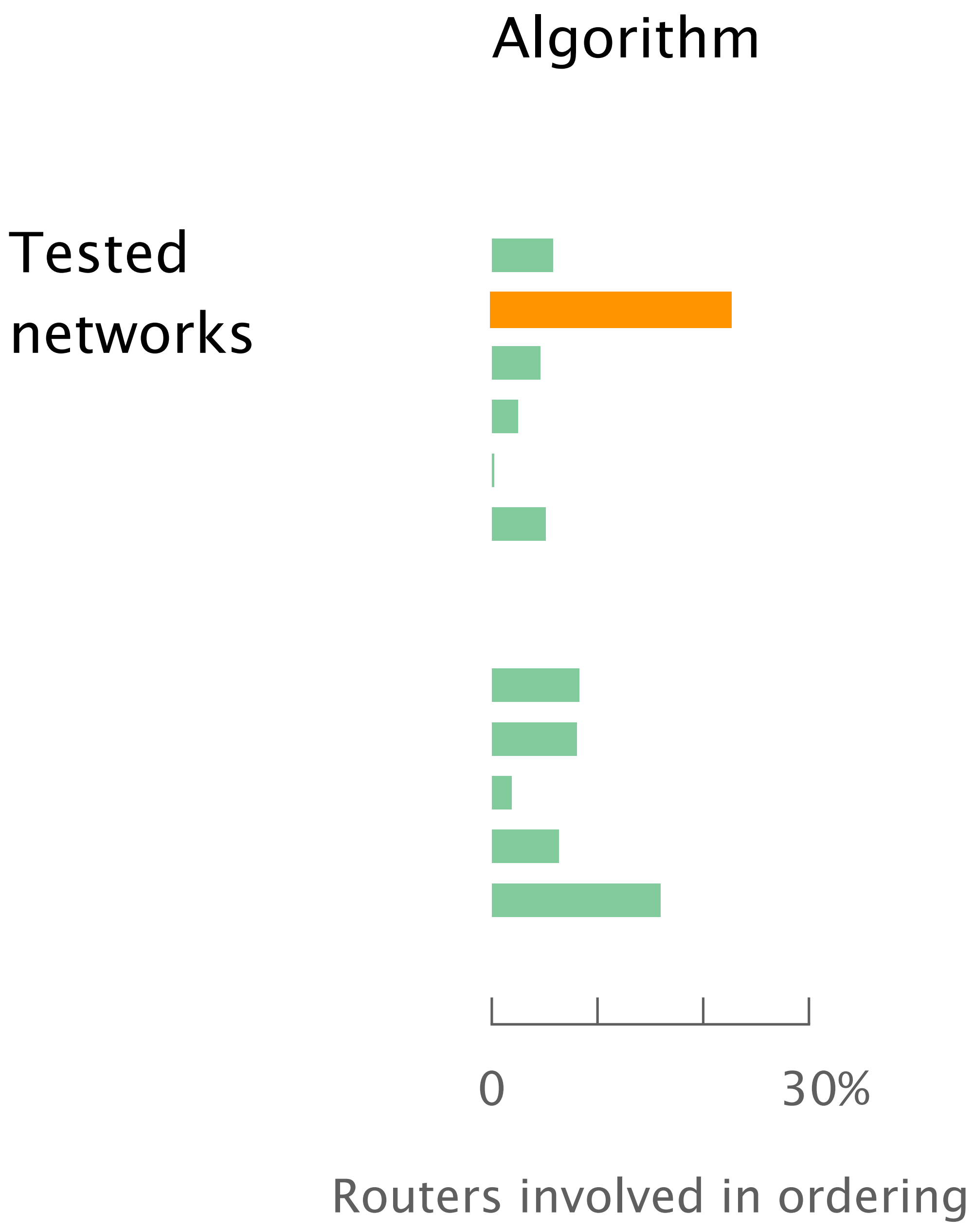
To avoid a forwarding loop towards LOSA,
KANS MUST be reconfigured before HOUS

An ordering does not always exist and deciding if one exists is hard

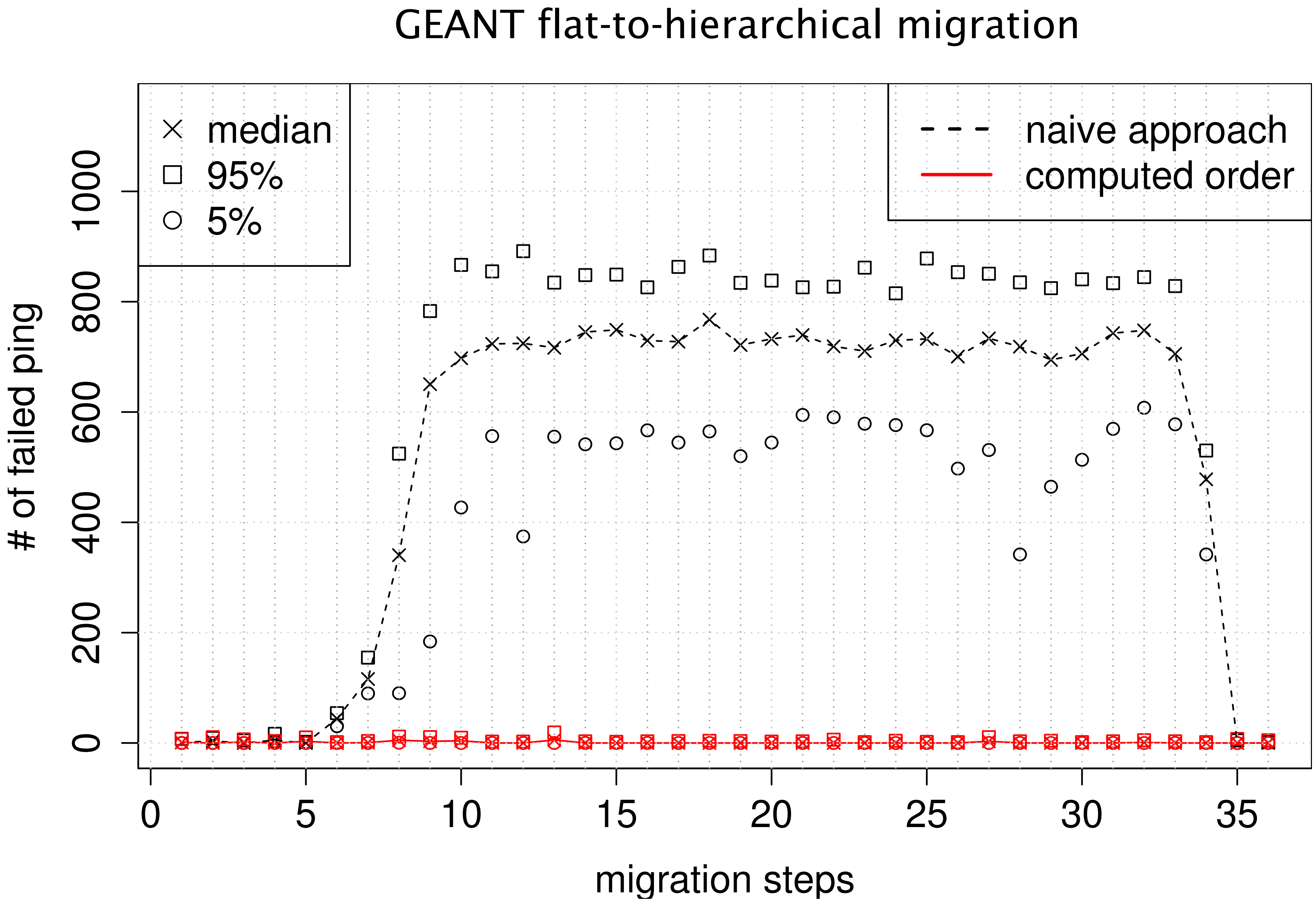
... but, in nearly all tested scenarios, the algorithm has found an ordering



More than 20% of the routers might be involved in the ordering

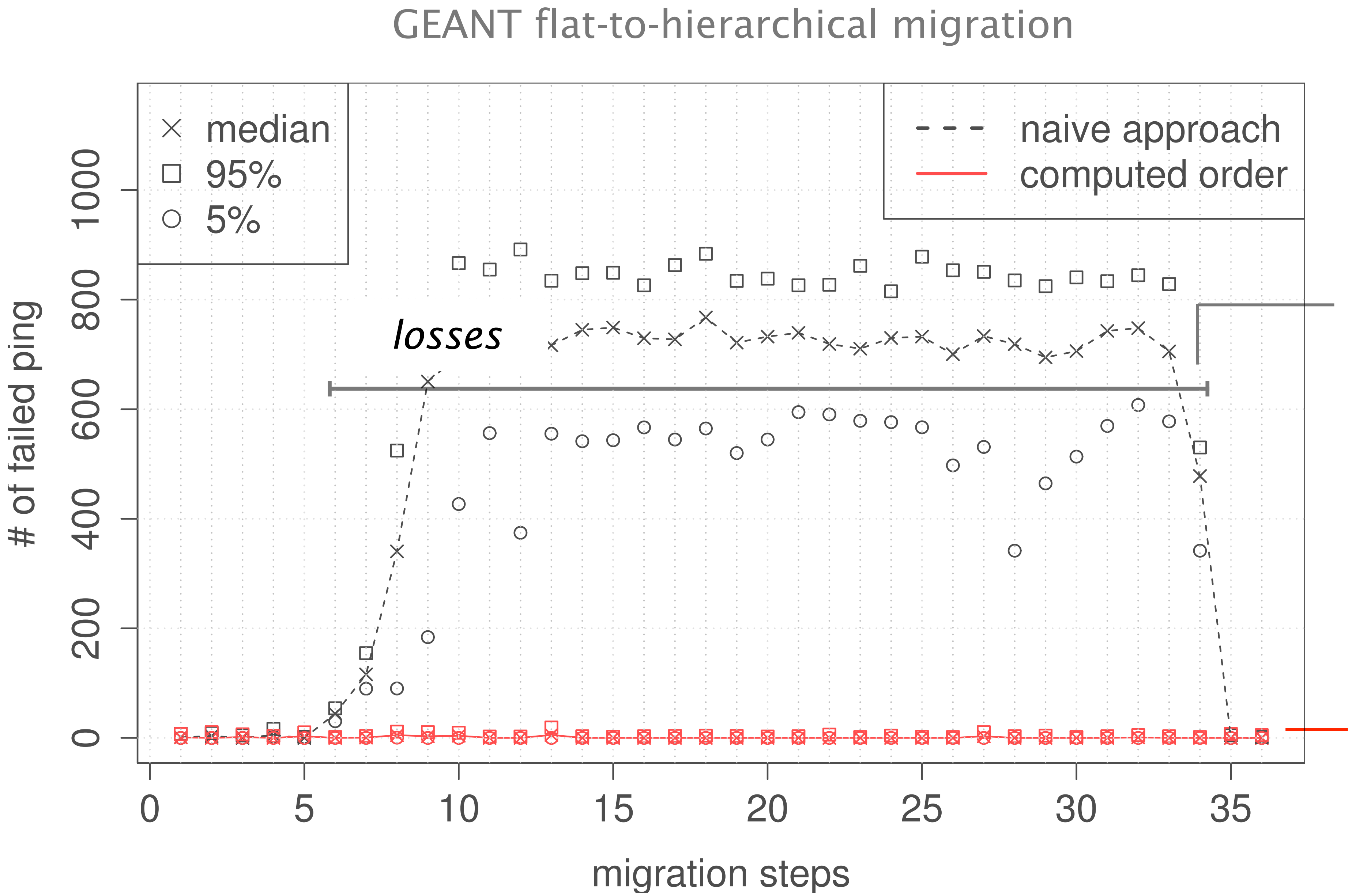


Using our ordering, we were able to achieve lossless reconfiguration



Average results (50 repetitions) computed on 700+ pings
per step from every router to 5 problematic destinations

By following the computed ordering, lossless IGP reconfiguration are possible



Traffic is lost for more than 80% of the process

No loss occurs with our framework

Average results (50 repetitions) computed on 700+ pings per step from every router to 5 problematic destinations

Methods and Techniques for Disruption-free Network Reconfiguration

Background

What is a network?

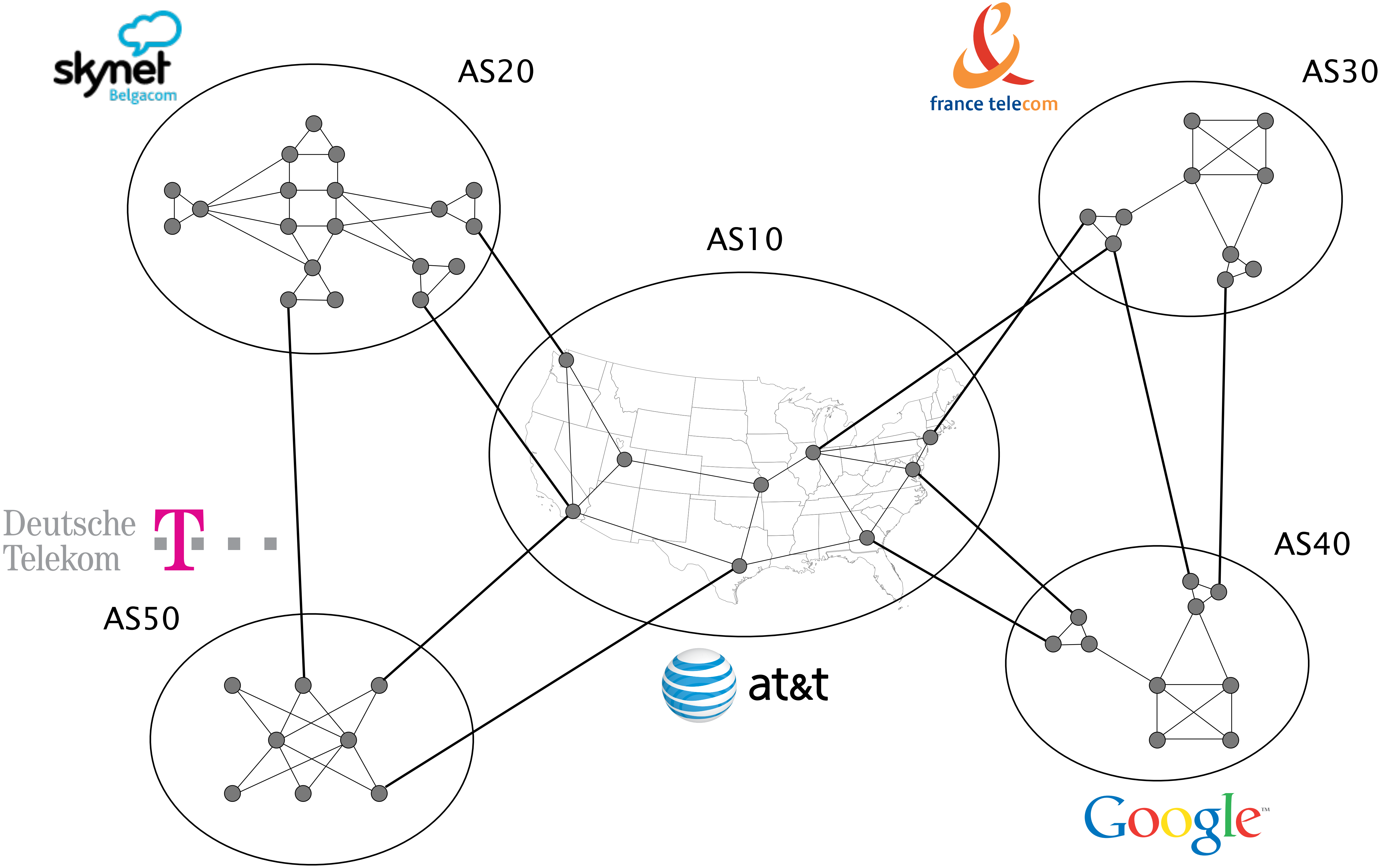
Intradomain reconfiguration

Find a reconfiguration ordering

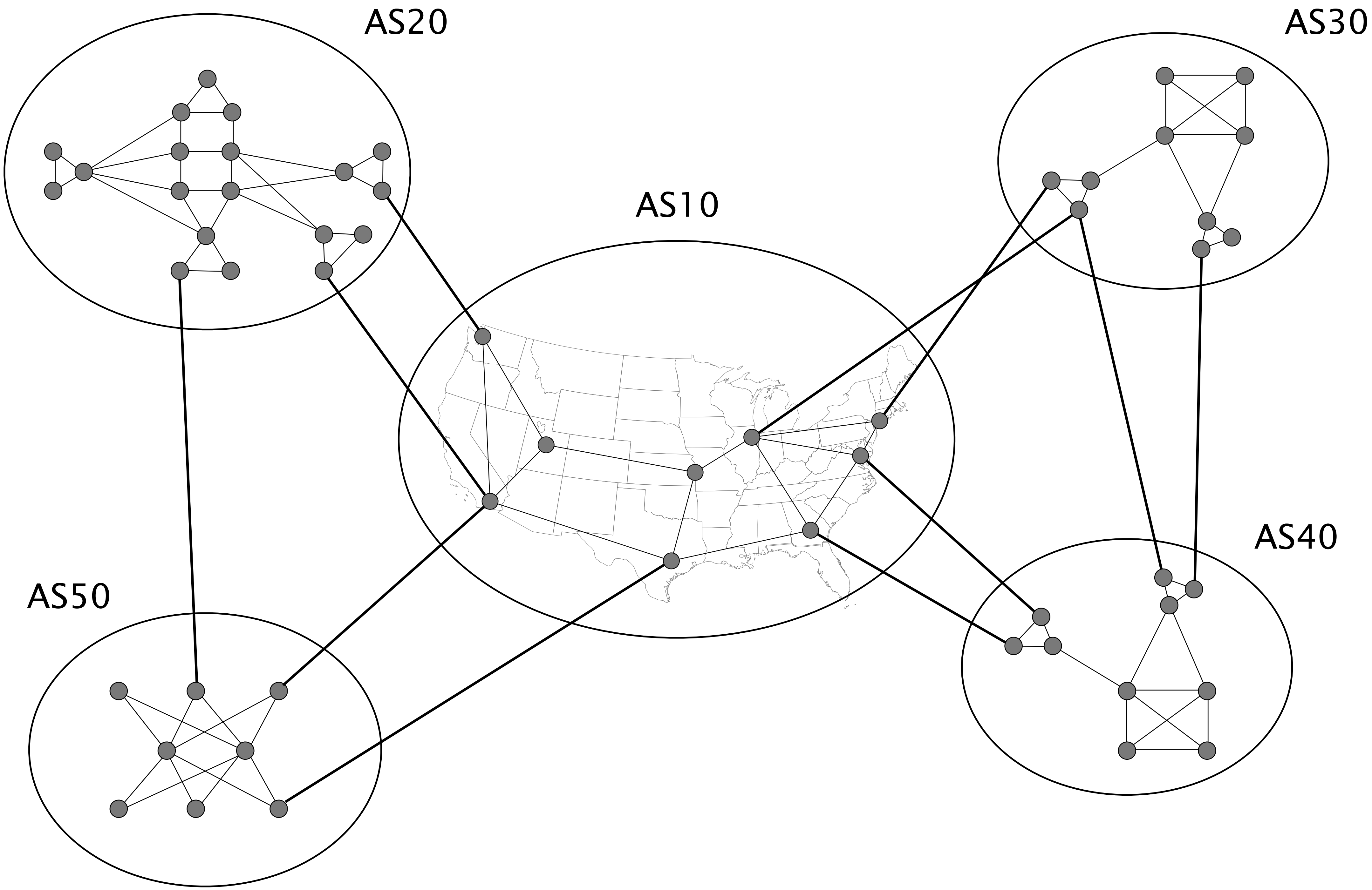
3 Interdomain reconfiguration

Overcome inherent complexity

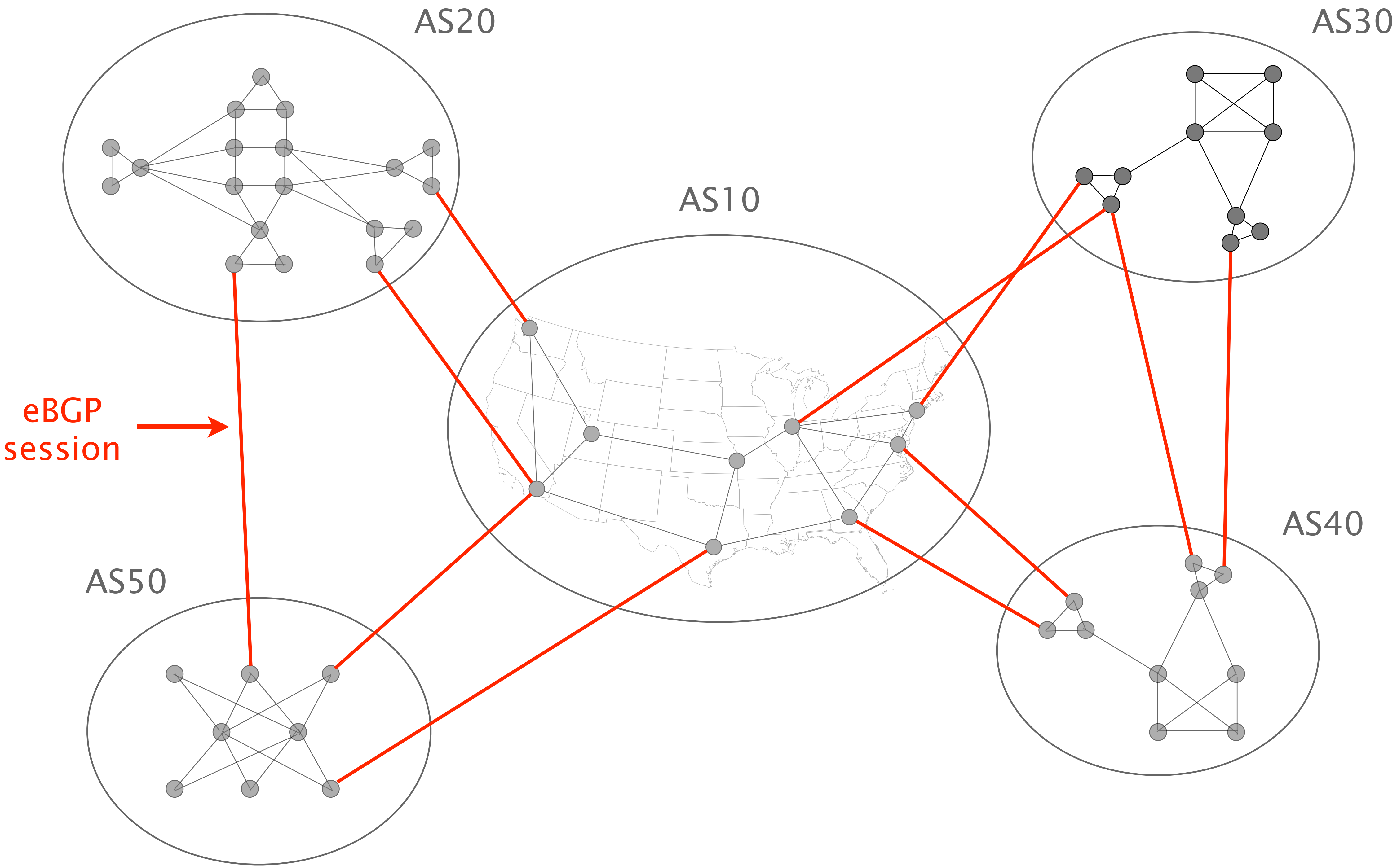
Interdomain routing protocols (BGP) rule traffic forwarding across routing domains



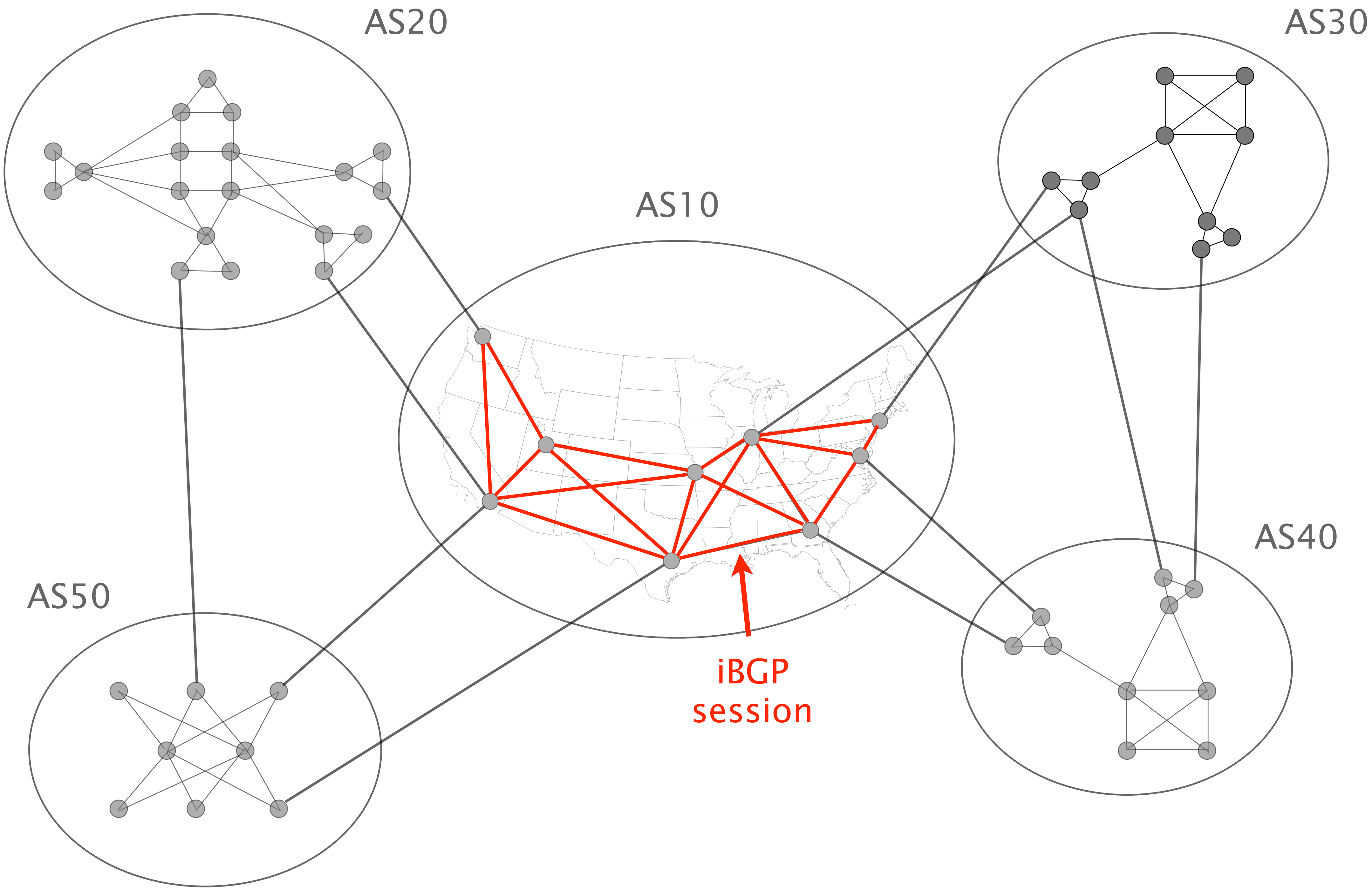
BGP comes in two flavors



external BGP (eBGP) exchanges reachability information between ASes



internal BGP (iBGP) distributes externally learned routes within the AS



Each flavor of a BGP configuration can be changed

Typical reconfiguration scenarios consist in

iBGP	Add sessions
	Remove sessions
	Change type (e.g., turn a router into a route-reflector)
eBGP	Add sessions
	Remove sessions
	Modify policies (e.g., turn a client into a peer)

Reconfiguring BGP can be disruptive

Reconfiguring BGP (*) can lead to

- routing oscillations [Griffin02]
- forwarding loops [Griffin02]
- blackholes [INFOCOM12]

or any combination of those

(*) [Guichard00, Smith10, Herrero10]

Reconfiguring BGP can be disruptive

Reconfiguring BGP (*) can lead to

- routing oscillations
- forwarding loops
- blackholes

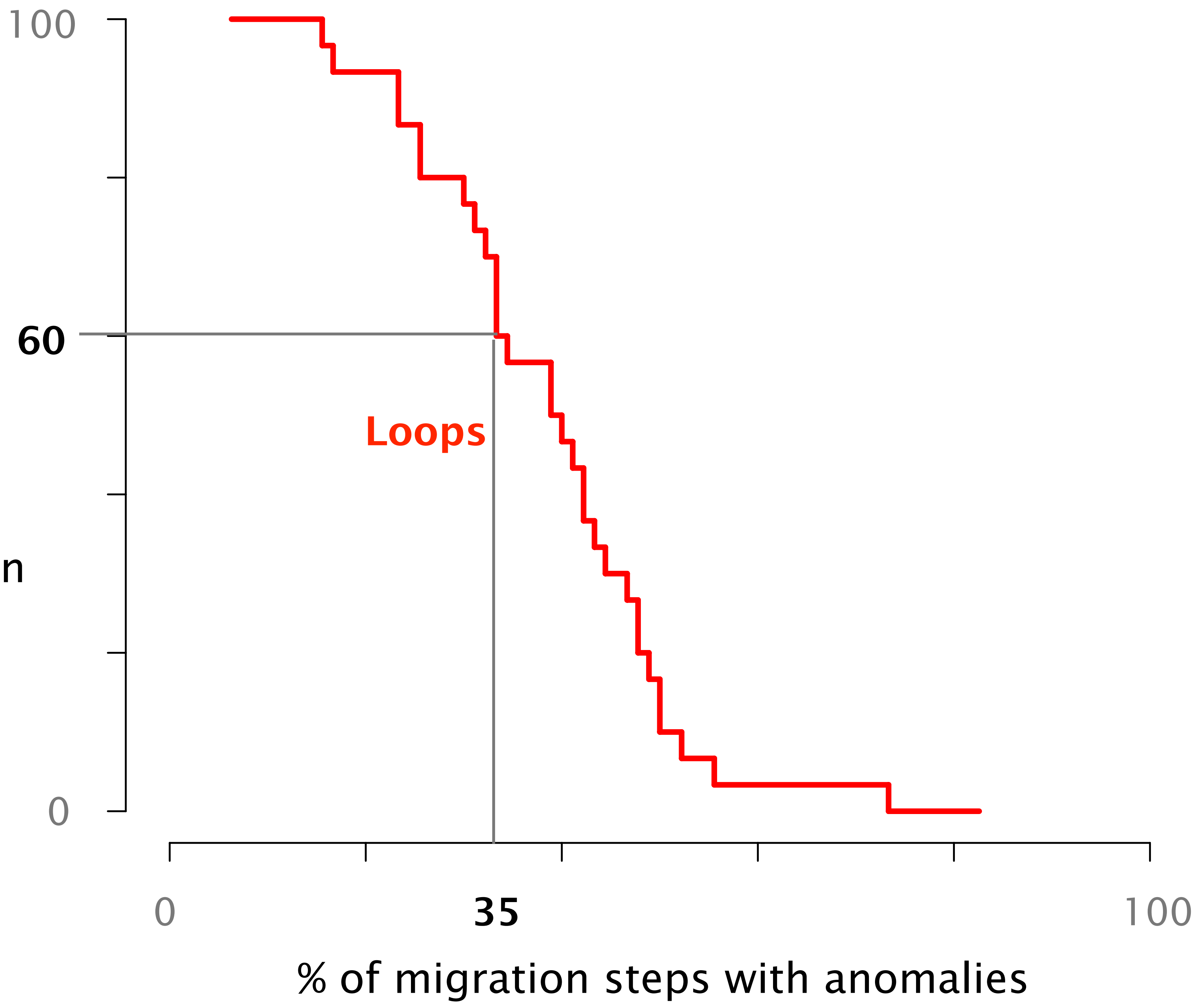
How many ?

or any combination of those

(*) [Guichard00, Smith10, Herrero10]

Best practices do not work

Tier1 experiments
(cumul. frequency)



60% of the experiments
were subject to loops
for > **35%** of the migration

Just like IGPs, finding an anomaly-free ordering is hard

Deciding if an anomaly-free ordering exists is at least NP-hard

It might even be harder

Just like IGPs, finding an anomaly-free ordering is hard and might not exist

Deciding if an anomaly-free ordering exists is at least NP-hard

It might even be harder

Due to contradictory constraint, anomaly-free ordering might not exist

Anomalies are guaranteed to appear, no matter what

But unlike IGPs, an algorithmic approach is not viable

There are way more BGP destinations than IGP ones

two orders of magnitude (i.e., 450.000 vs 1000s)

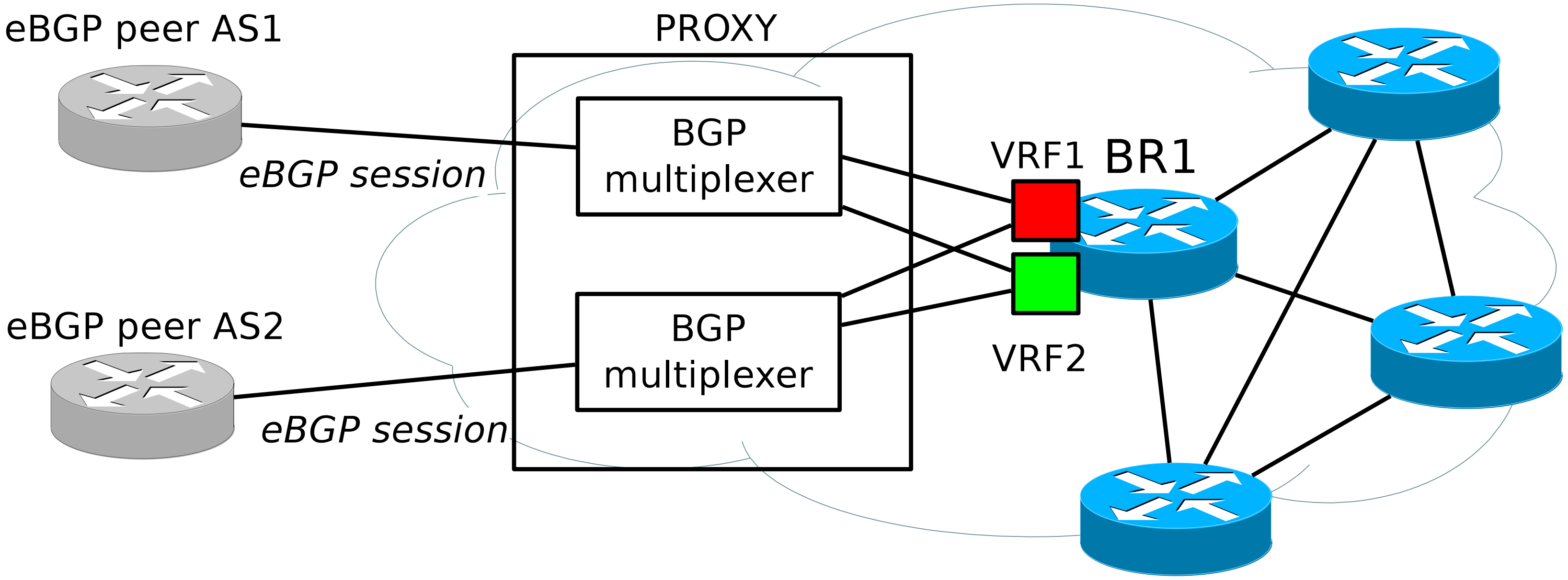
BGP destinations can be announced from any subset of nodes

while IGP destinations are usually announced from 1 node

Local changes can have remote impact

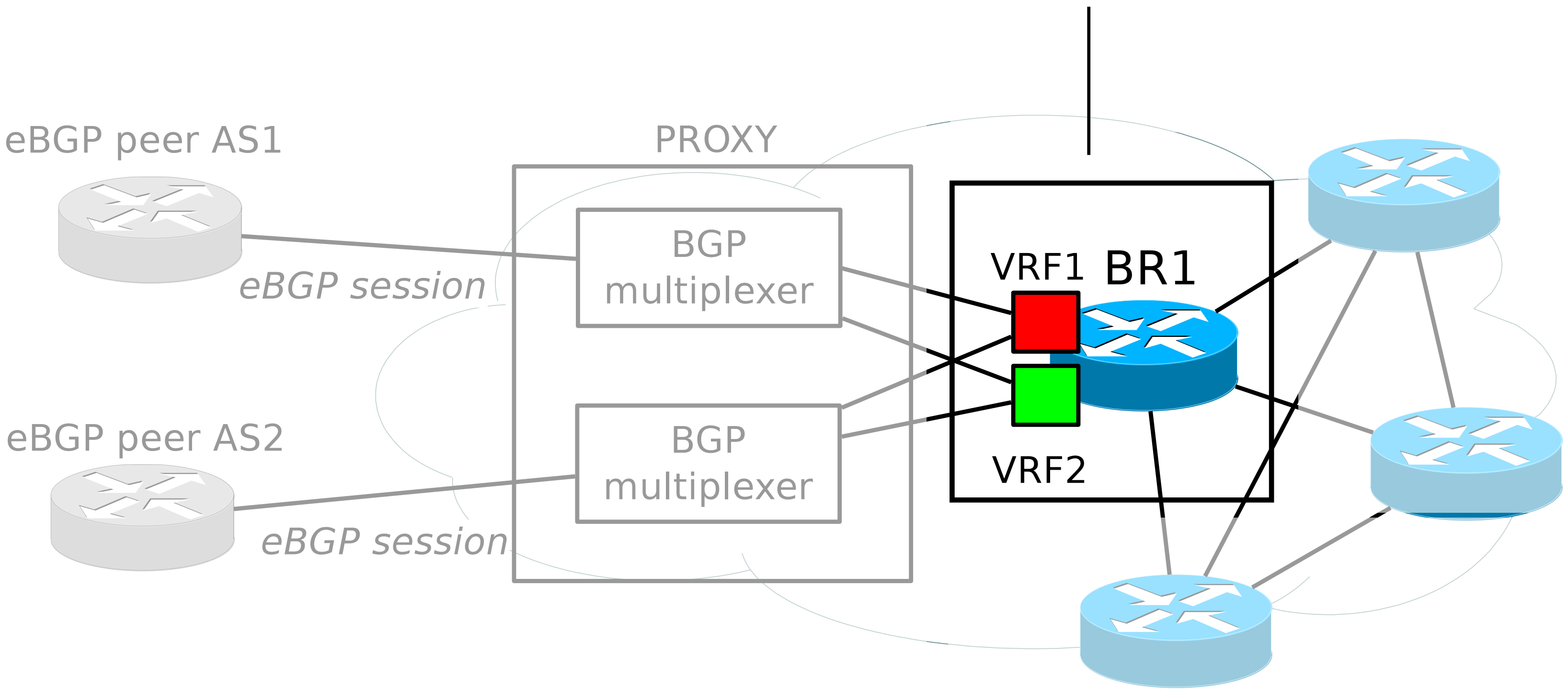
meaning we must them into account as well

To circumvent the inherent complexity, we developed a reconfiguration framework



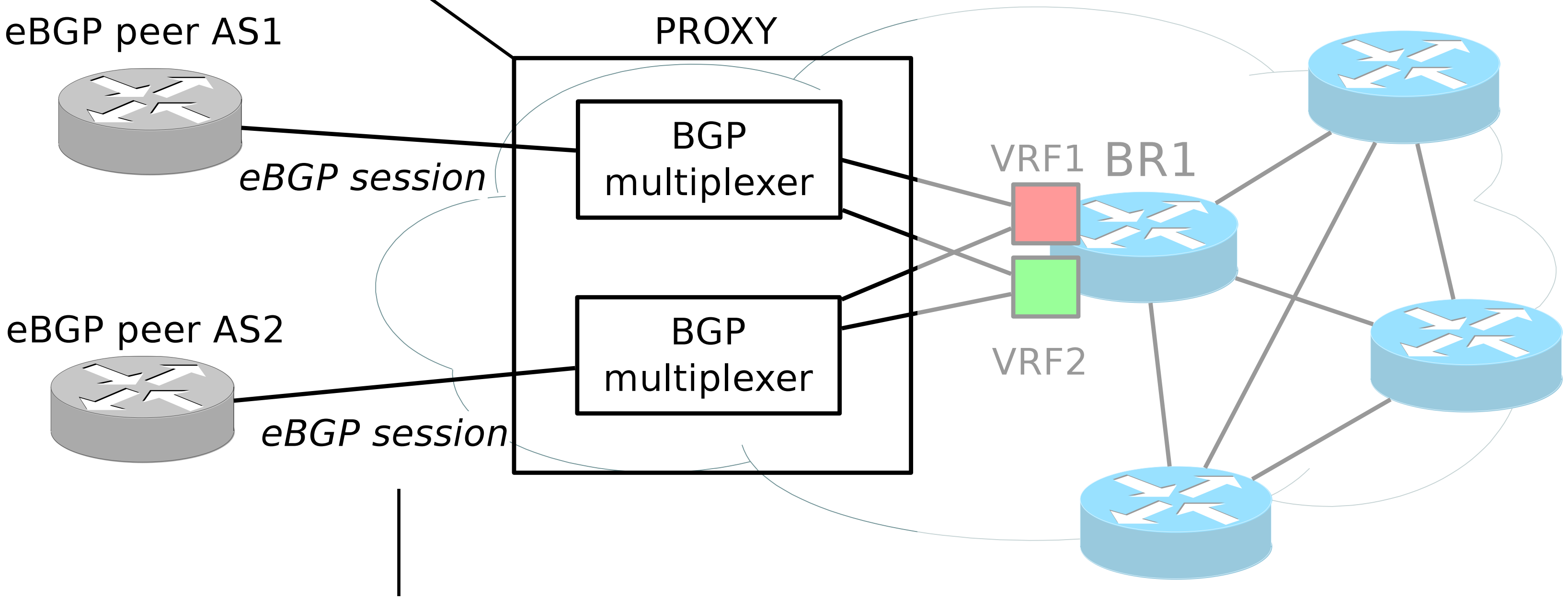
To circumvent the inherent complexity, we developed a reconfiguration framework

By leveraging specific technologies (L3VPNs), routers can maintain different BGP routing planes



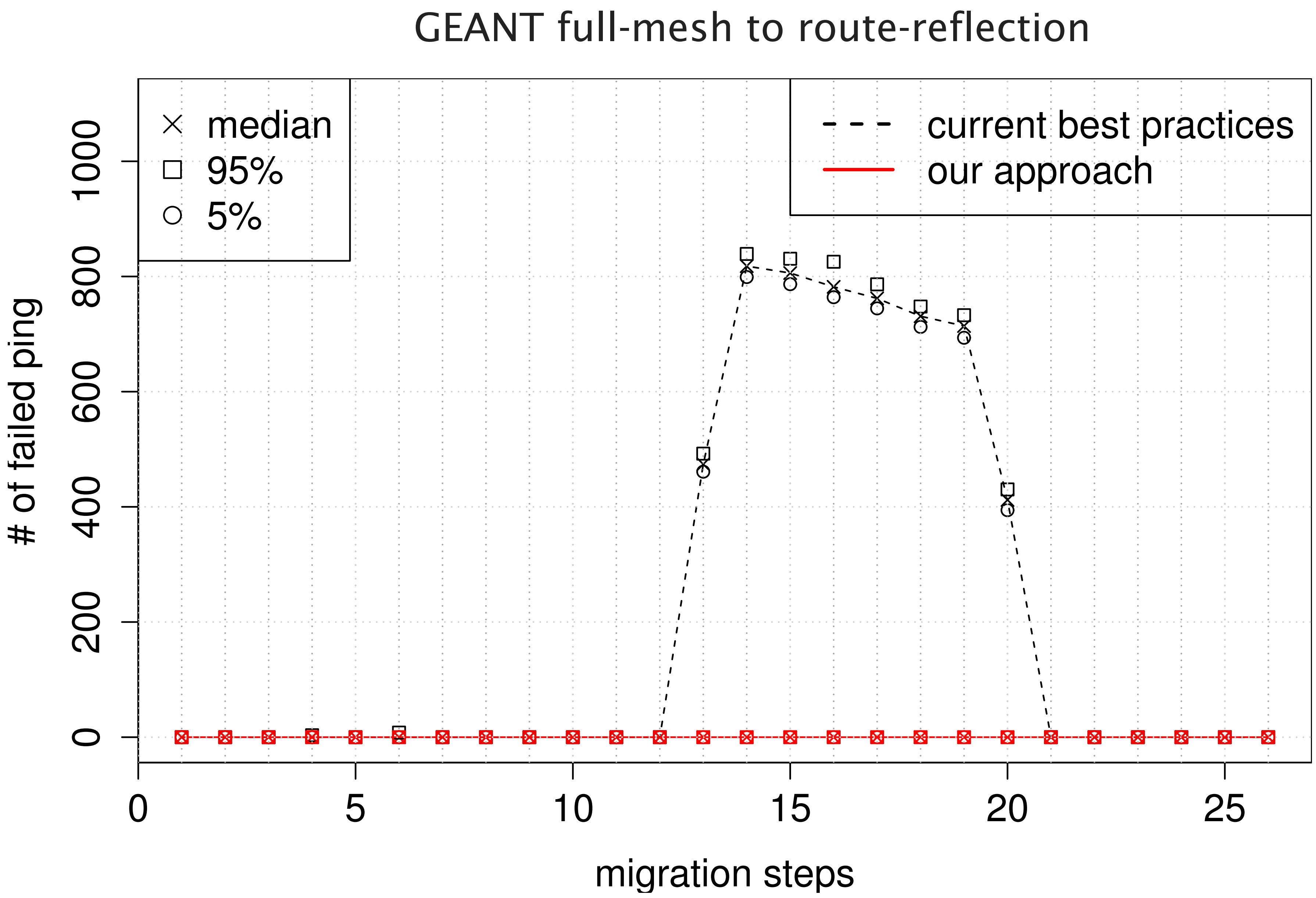
To circumvent the inherent complexity, we developed a reconfiguration framework

A proxy distributes BGP updates to the different routing planes



Our framework is completely transparent for neighboring router

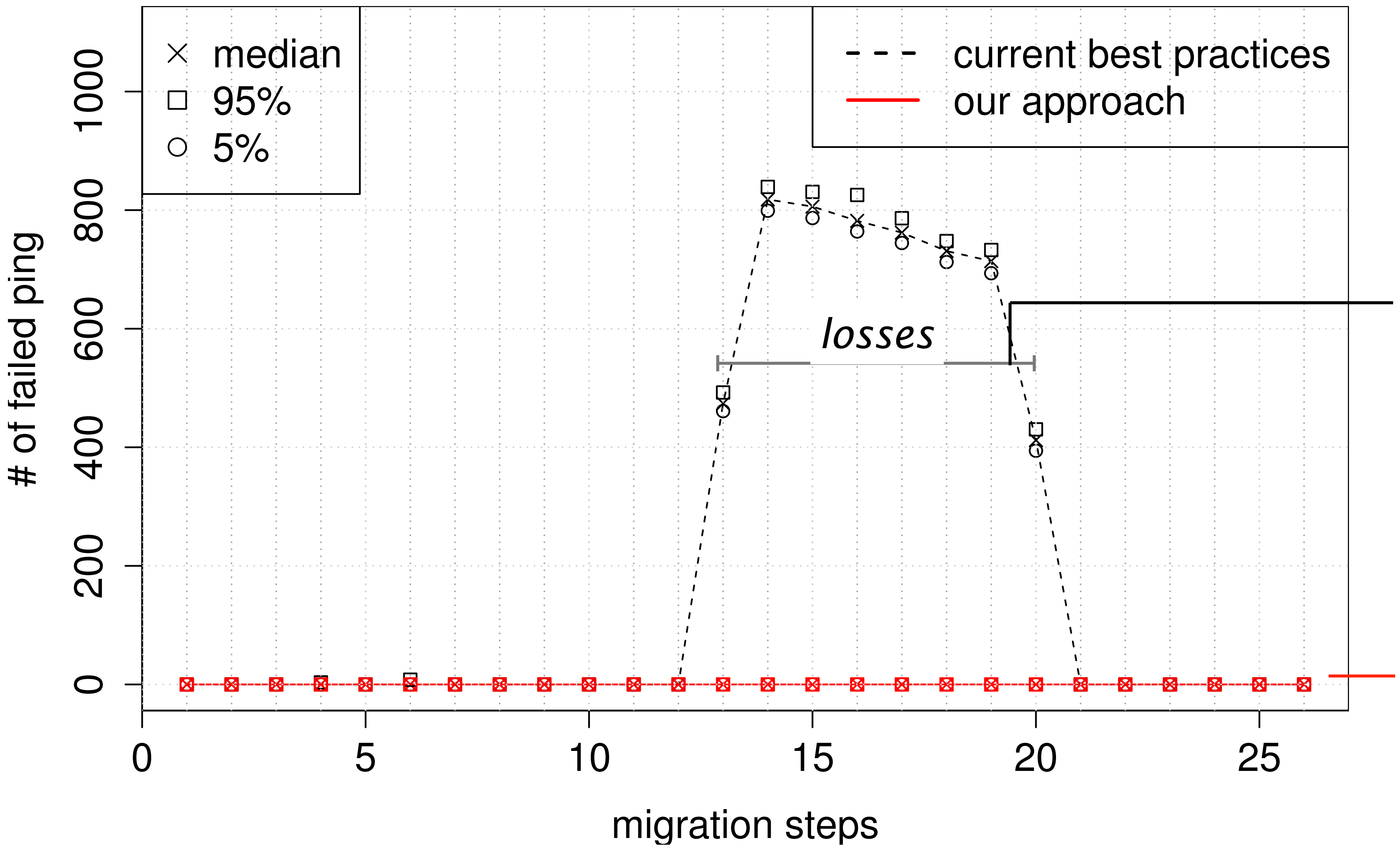
Our reconfiguration framework enables lossless reconfiguration



Average results (30 repetitions) computed on 120+ pings per step from every router to 16 summary prefixes

Our reconfiguration framework enables lossless reconfiguration

GEANT full-mesh to route-reflection



losses from 7 routers

60% of GEANT routing table is impacted

No loss occurs with our framework

Average results (30 repetitions) computed on 120+ pings per step from every router to 16 summary prefixes

Methods and Techniques for Disruption-free Network Reconfiguration

Background

What is a network?

Intradomain reconfiguration

Find a reconfiguration ordering

Interdomain reconfiguration

Overcome inherent complexity

Progressively reconfigure a running network
without creating any anomaly

High-level overview of the contributions

Provide a deep theoretical and practical understanding of routing reconfiguration problems

Bring flexibility to network management

regularly move to the best network-wide configuration

Development of a complete reconfiguration framework

which works in today’s networks

Publications

Part I

[SIGCOMM11] Laurent Vanbever, Stefano Vissicchio, Cristel Pelsser, Pierre Francois and Olivier Bonaventure. Seamless Network-Wide IGP Migrations. In *ACM SIGCOMM Conference*, 2011

[TON12a] Laurent Vanbever, Stefano Vissicchio, Cristel Pelsser, Pierre Francois and Olivier Bonaventure. Lossless Migrations of Link-State IGPs. In *IEEE/ACM Transactions on Networking*, 2012. (To appear).

Part II

[INFOCOM12] Stefano Vissicchio, Luca Cittadini, Laurent Vanbever and Olivier Bonaventure. iBGP Deceptions: More Sessions, Fewer Routes. In *IEEE INFOCOM*, 2012

[TON12b] Stefano Vissicchio, Laurent Vanbever, Cristel Pelsser, Luca Cittadini, Pierre Francois and Olivier Bonaventure. Improving Network Agility with Seamless BGP Reconfigurations. In *IEEE/ACM Transactions on Networking*, 2012. (To appear).

Publications

Part III

[INFOCOM13?] Laurent Vanbever, Stefano Vissichio, Luca Cittadini, and Olivier Bonaventure. When the Cure is Worse than the Disease: the Impact of Graceful IGP Operations on BGP. Submitted to IEEE INFOCOM, 2013

Part IV

[INM08] Laurent Vanbever, Grégory Pardoen and Olivier Bonaventure. Towards Validated Network Configurations with NCGuard. In Proc. of Internet Network Management Workshop, 2008

[PRESTO09] Laurent Vanbever, Bruno Quoitin and Olivier Bonaventure. A Hierarchical Model for BGP Routing Policies. In Proc. of the Second ACM SIGCOMM Workshop on Programmable Routers for Extensible Services of TOmorrow, 2009.