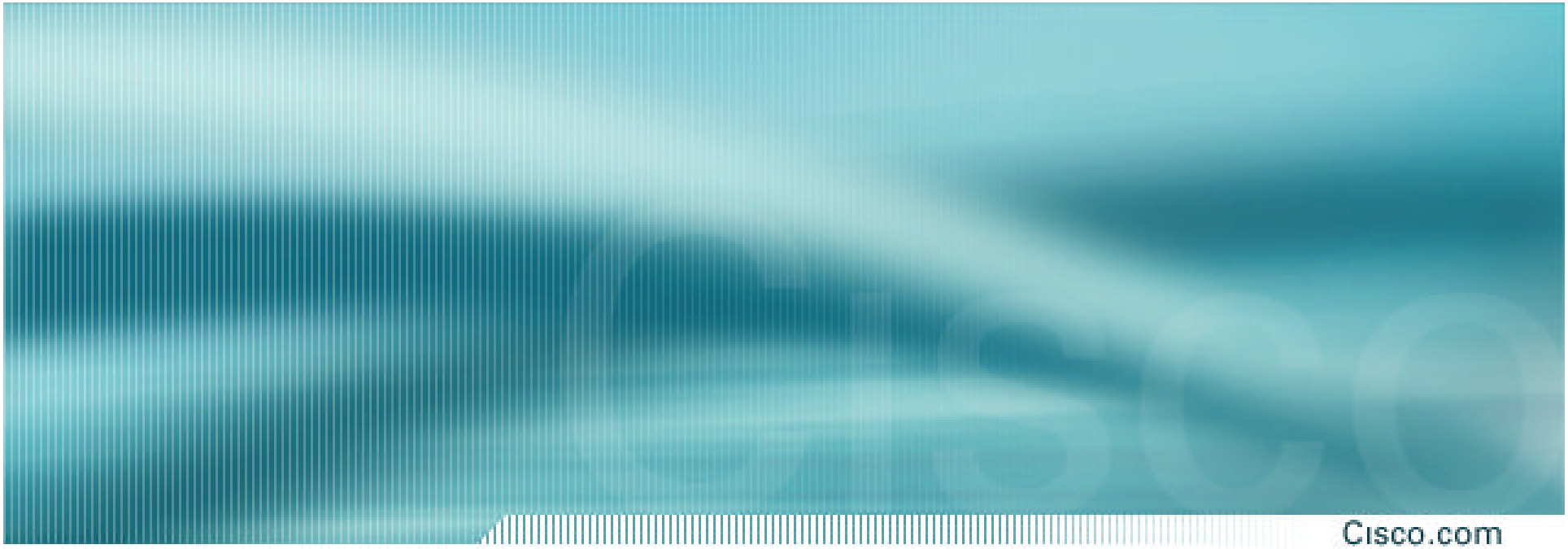


BGP and the Internet

Multihoming

Multihoming Definition

- More than one link external to the local network
  - two or more links to the same ISP
  - two or more links to different ISPs
- Usually two external facing routers
  - one router gives link and provider redundancy only

Multihoming

- **The scenarios described here apply equally well to end sites being customers of ISPs and ISPs being customers of other ISPs**
- **Implementation detail may be different**

end site ® ISP	**ISP controls config**
ISP1 ® ISP2	**ISPs share config**

AS Numbers

- An Autonomous System Number is required by BGP
- Obtained from upstream ISP or Regional Registry
- Necessary when you have links to more than one ISP or exchange point

Configuring Policy

- **Assumptions:**
  - **prefix-lists are used throughout**
  - **easier/better/faster than access-lists**
- **Three BASIC Principles**
  - **prefix-lists to filter prefixes**
  - **filter-lists to filter ASNs**
  - **route-maps to apply policy**

Policy Tools

- **Local preference**
  - **outbound traffic flows**
- **Metric (MED)**
  - **inbound traffic flows (local scope)**
- **AS-PATH prepend**
  - **inbound traffic flows (Internet scope)**
- **Communities**
  - **specific inter-provider peering**

Originating Prefixes

- **Basic Assumptions**

  MUST announce assigned address block to Internet

  MAY also announce subprefixes – reachability is not guaranteed

  RIR minimum allocation is /20

    several ISPs filter RIR blocks on this boundary

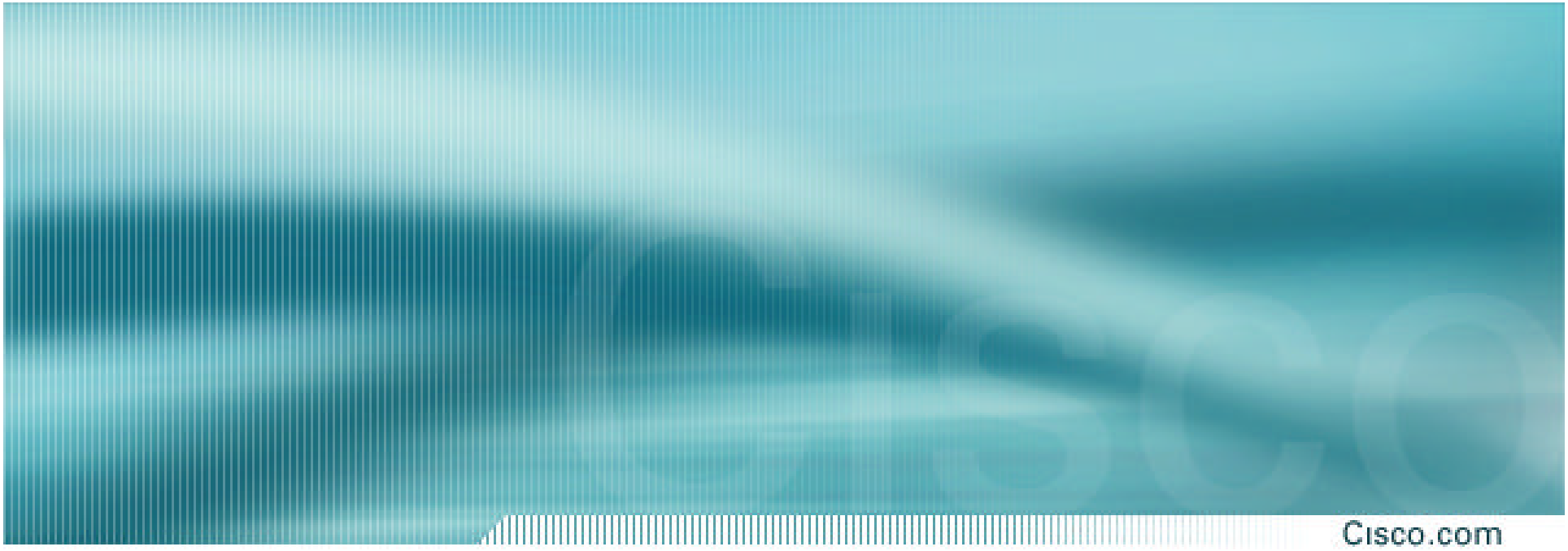
    called “Net Police” filters by some

Part of the “Net Police” prefix list


```
!! APNIC
ip prefix-list FILTER permit 61.0.0.0/8 ge 9 le 20
ip prefix-list FILTER permit 202.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 210.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 218.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 220.0.0.0/7 ge 9 le 20
!! ARIN
ip prefix-list FILTER permit 24.0.0.0/8 ge 9 le 20
ip prefix-list FILTER permit 63.0.0.0/8 ge 9 le 20
ip prefix-list FILTER permit 64.0.0.0/6 ge 9 le 20
ip prefix-list FILTER permit 68.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 199.0.0.0/8 ge 9 le 20
ip prefix-list FILTER permit 204.0.0.0/6 ge 9 le 20
ip prefix-list FILTER permit 208.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 216.0.0.0/8 ge 9 le 20
!! RIPE NCC
ip prefix-list FILTER permit 62.0.0.0/8 ge 9 le 20
ip prefix-list FILTER permit 80.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 82.0.0.0/8 ge 9 le 20
ip prefix-list FILTER permit 193.0.0.0/8 ge 9 le 20
ip prefix-list FILTER permit 194.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 212.0.0.0/7 ge 9 le 20
ip prefix-list FILTER permit 217.0.0.0/8 ge 9 le 20
!! LACNIC
ip prefix-list FILTER permit 200.0.0.0/8 ge 9 le 20
```

“Net Police” prefix list issues

- meant to “punish” ISPs who pollute the routing table with specifics rather than announcing aggregates
- impacts legitimate multihoming especially at the Internet’s edge
- impacts regions where domestic backbone is unavailable or costs $$$ compared with international bandwidth
- hard to maintain – requires updating when RIRs start allocating from new address blocks
- don’t do it unless consequences understood and you are prepared to keep the list current

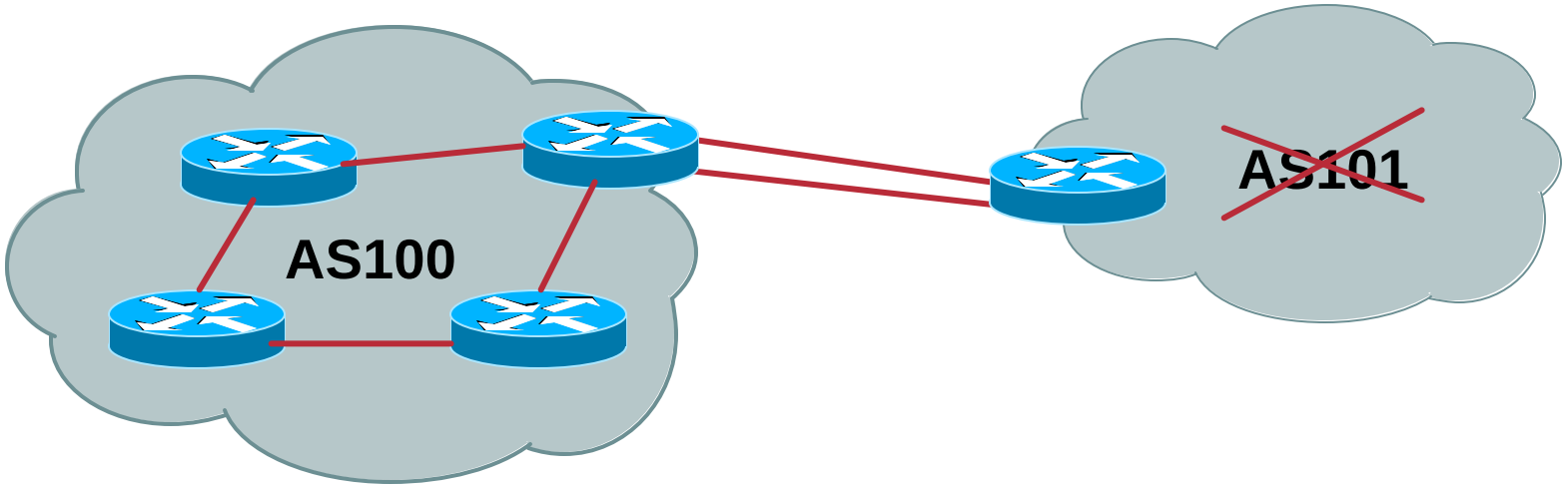


Multihoming Options

Multihoming Scenarios

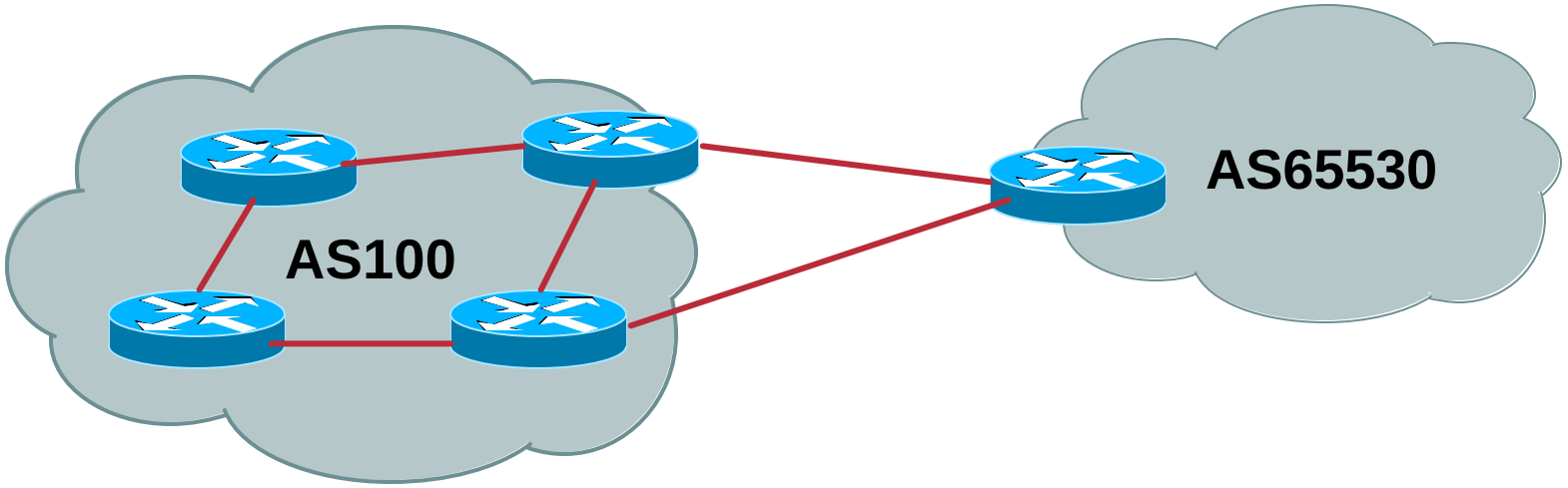
- **Stub network**
- **Multi-homed stub network**
- **Multi-homed network**
- **Configuration Options**

Stub Network



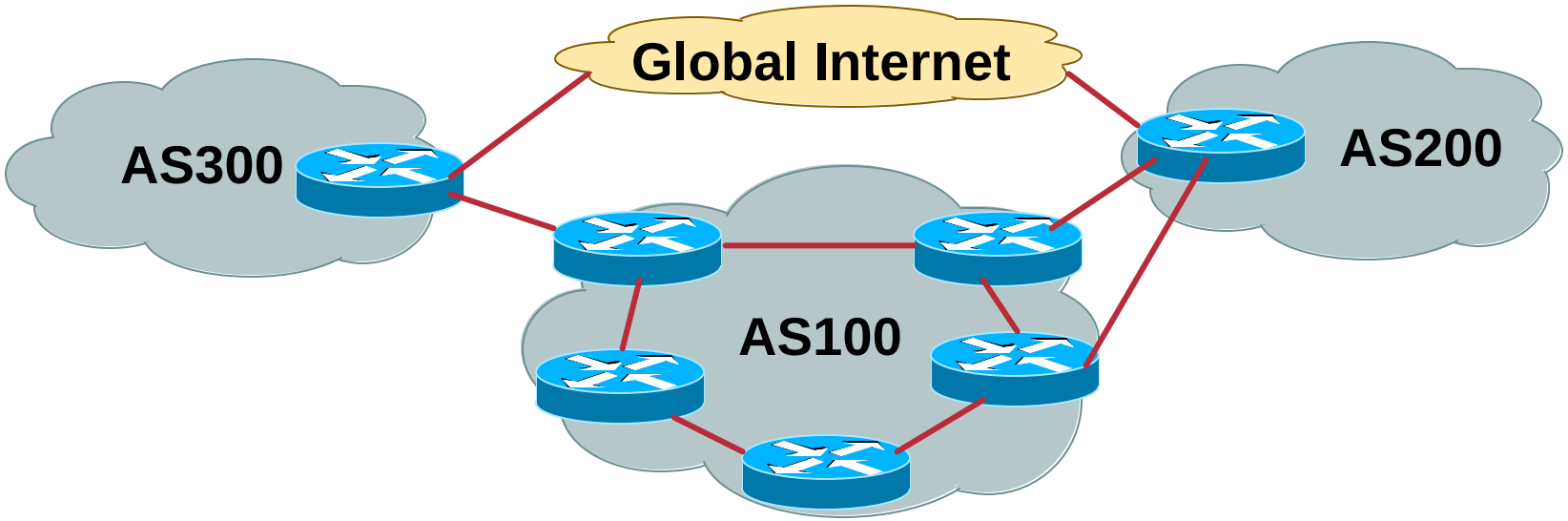
- No need for BGP
- Point static default to upstream ISP
- Upstream ISP advertises stub network
- Policy confined within upstream ISP's policy

Multi-homed Stub Network



- Use BGP (not IGP or static) to loadshare
- Use private AS (ASN > 64511)
- Upstream ISP advertises stub network
- Policy confined within upstream ISP's policy

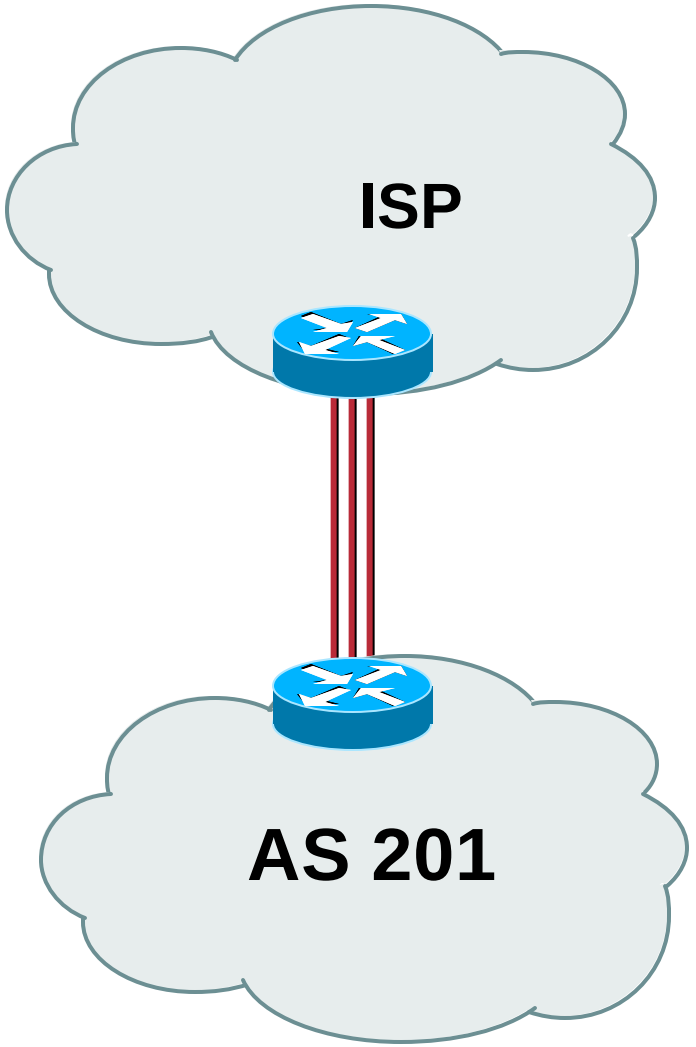
Multi-Homed Network



- **Many situations possible**
  - **multiple sessions to same ISP**
  - **secondary for backup only**
  - **load-share between primary and secondary**
  - **selectively use different ISPs**

Multiple Sessions to an ISP

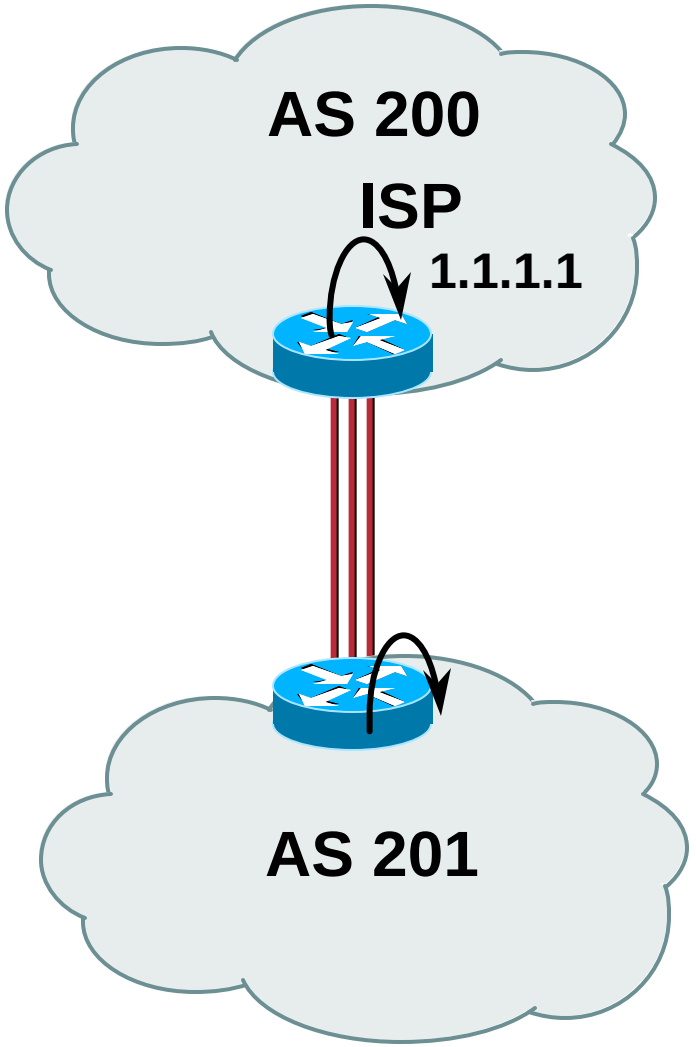
- **Several options**
  - **ebgp multihop**
  - **bgp multipath**
  - **cef loadsharing**
  - **bgp attribute manipulation**



Multiple Sessions to an ISP ebgp multihop

- eBGP to loopback addresses
- eBGP prefixes learned with loopback address as next hop

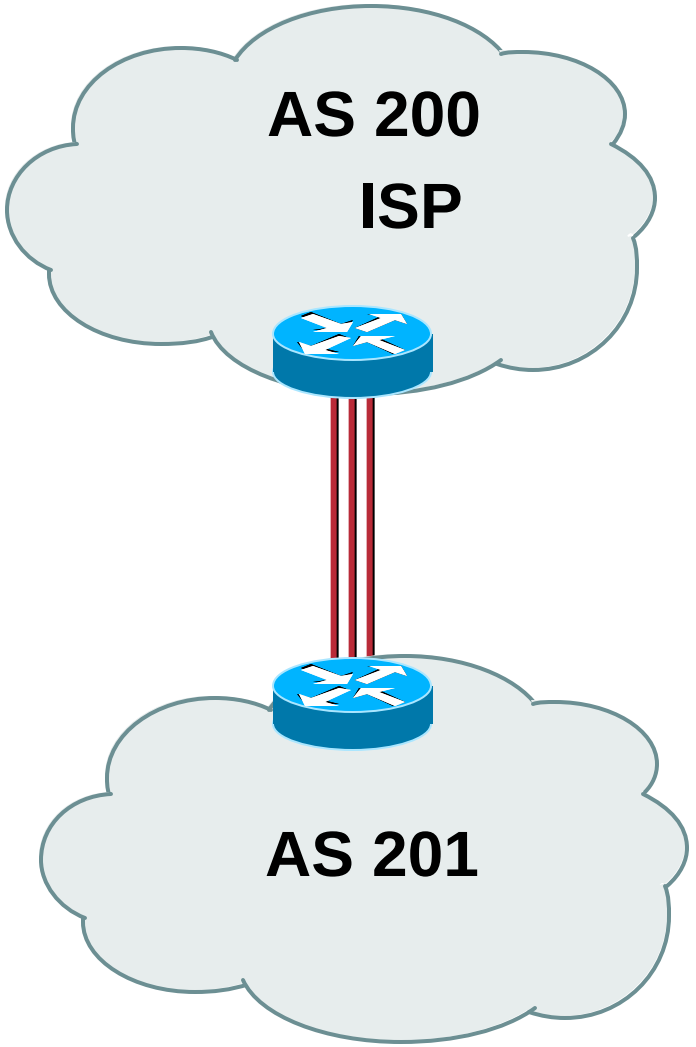
```
router bgp 201
 neighbor 1.1.1.1 remote-as 200
 neighbor 1.1.1.1 ebgp-multihop 2
ip route 1.1.1.1 255.255.255.255 serial 1/0
ip route 1.1.1.1 255.255.255.255 serial 1/1
ip route 1.1.1.1 255.255.255.255 serial 1/2
```



Multiple Sessions to an ISP bgp multi path

- Three BGP sessions required
- limit of 6 parallel paths

```
router bgp 201
 neighbor 1.1.2.1 remote-as 200
 neighbor 1.1.2.5 remote-as 200
 neighbor 1.1.2.9 remote-as 200
 maximum-paths 3
```



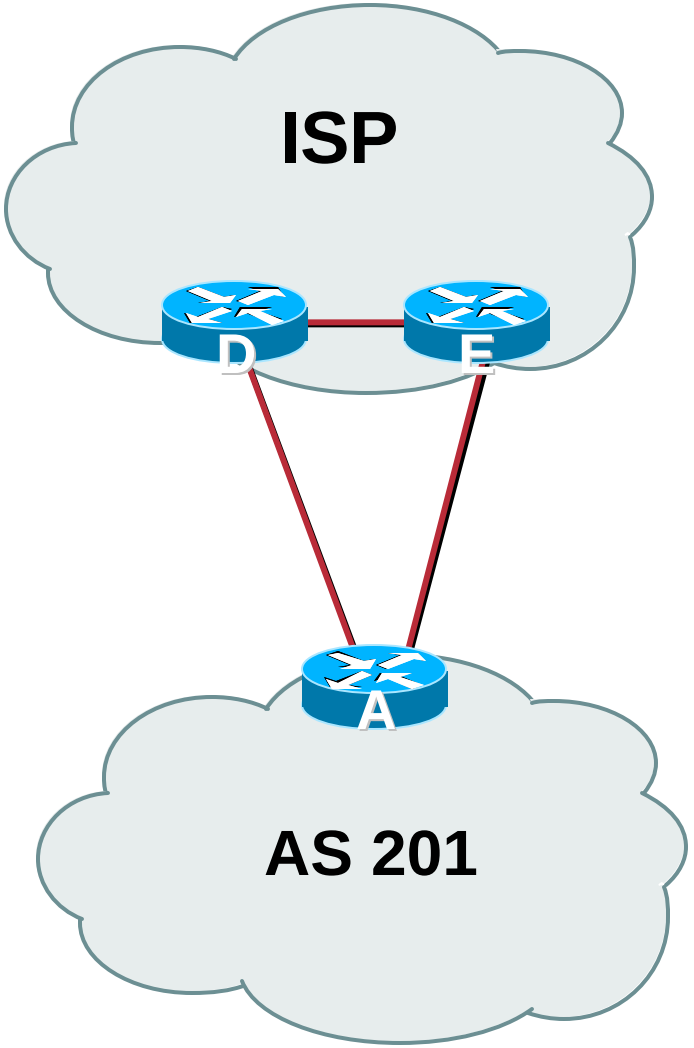
Multiple Sessions to an ISP

- **Use eBGP multi-path to install multiple paths in IP table**

```
router bgp 201
 maximum-path <1-6>
```

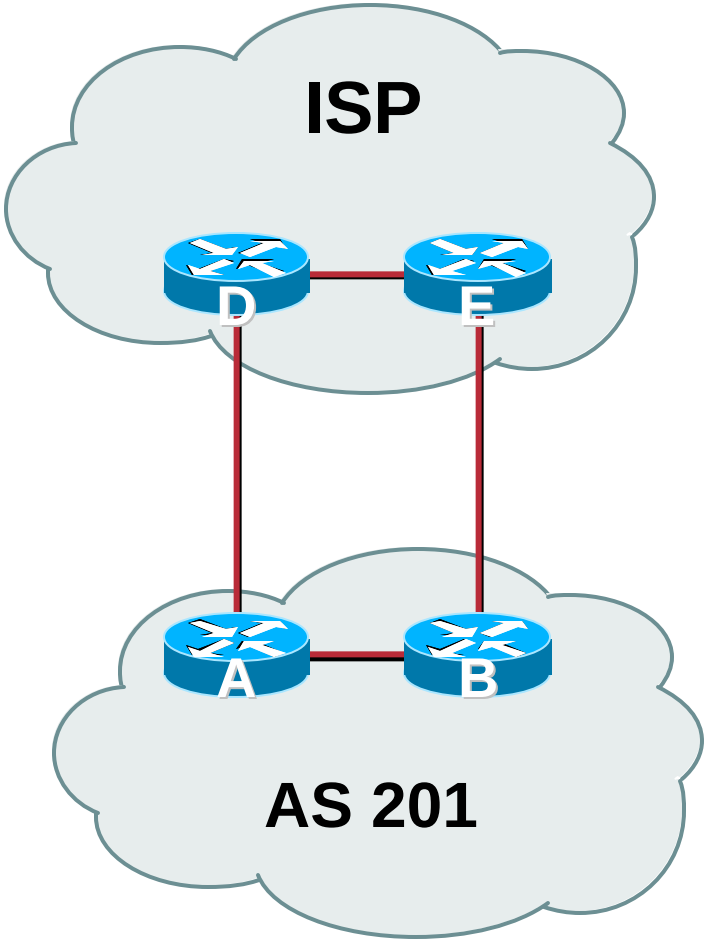
- **Load share over the alternate paths**

  per destination loadsharing



Multiple Sessions to an ISP

- **Simplest scheme is to use defaults**
- **Learn/advertise prefixes for better control**



Multiple Sessions to ISPs

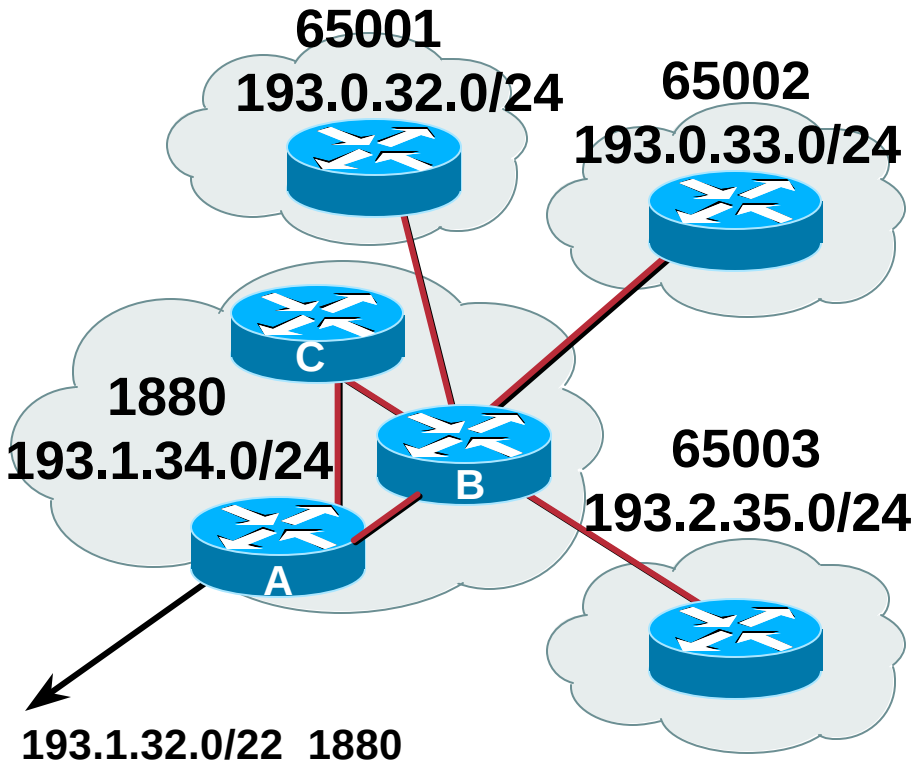
- **Planning and some work required to achieve load sharing**
  - **Point default towards one ISP**
  - **Learn selected prefixes from second ISP**
  - **Modify the number of prefixes learnt to achieve acceptable load sharing**
- **No magic solution**

Private-AS – Application

- **Applications**

  ISP with single-homed customers

  corporate network with several regions and connections to the Internet only in the core



Private-AS Removal

- **neighbor x.x.x.x remove-private-AS**
- **Rules:**

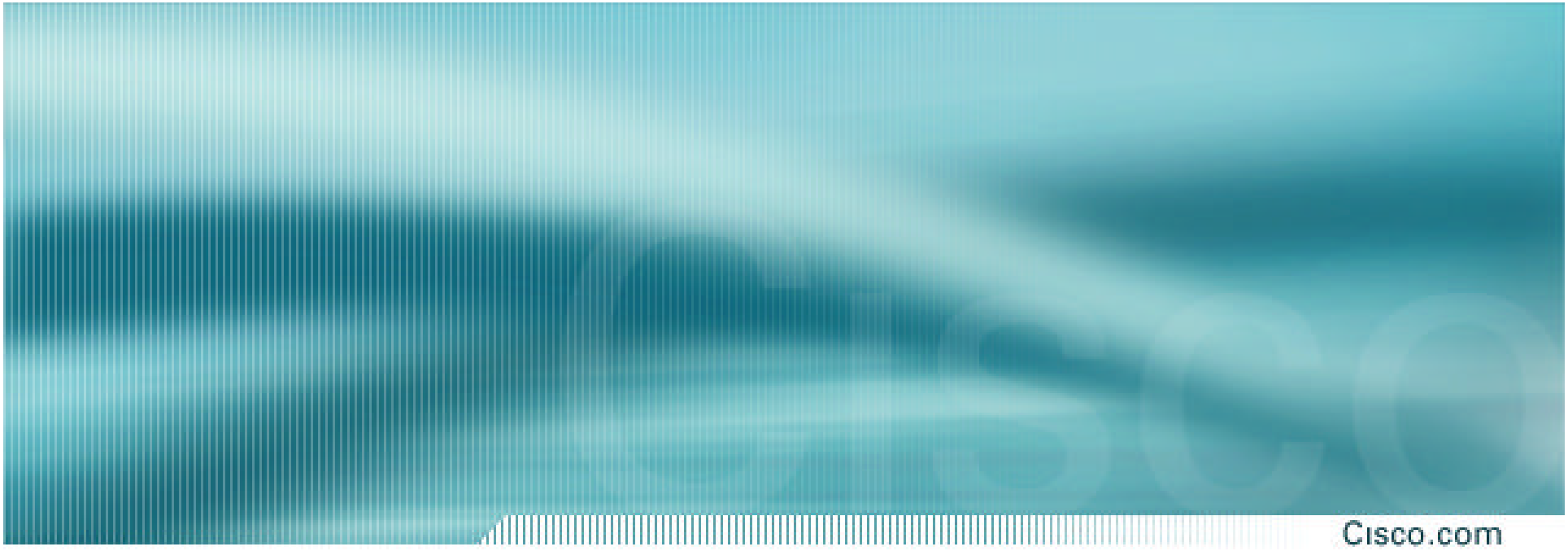
  available for eBGP neighbors only

  if the update has AS_PATH made up of private-AS numbers, the private-AS will be dropped

  if the AS_PATH includes private and public AS numbers, private AS number will not be removed…it is a configuration error!

  if AS_PATH contains the AS number of the eBGP neighbor, the private-AS numbers will not be removed

  if used with confederations, it will work as long as the private AS numbers are after the confederation portion of the AS_PATH



Two links to the same ISP

Basic – No Redundancy

Two links to the same ISP

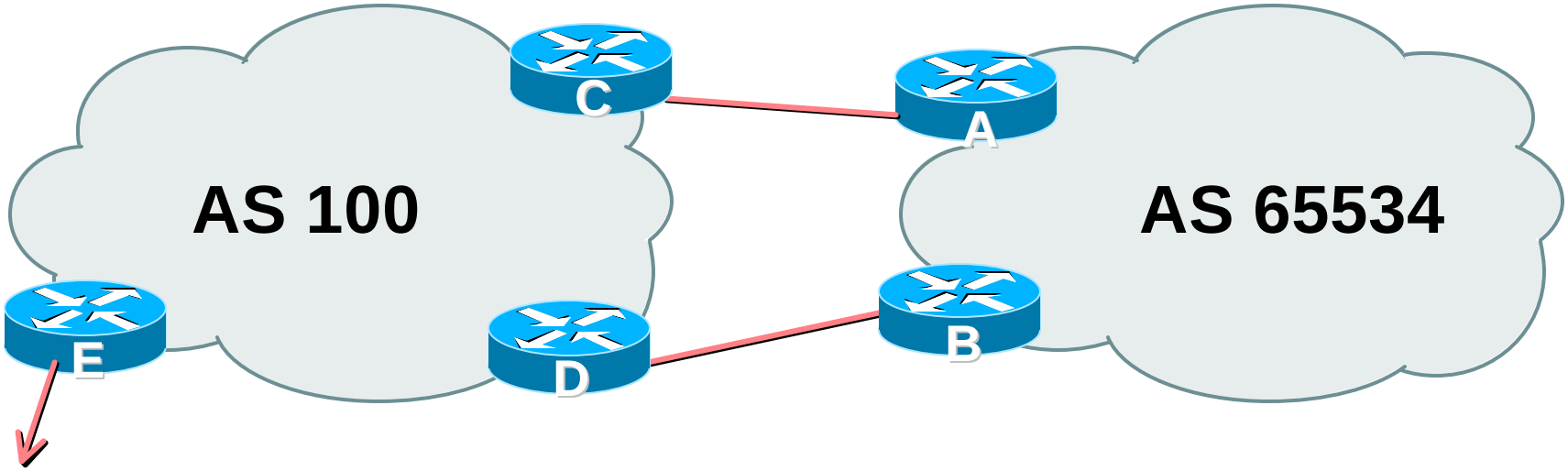
- **Can use BGP for this to aid loadsharing**

  use a private AS (ASN > 64511)

- **upstream ISP proxy aggregates**

  in other words, announces only your address block to the Internet (as would be done if you had one statically routed connection)

Two links to the same ISP



- **AS100 proxy aggregates for AS 65534**

Two links to the same ISP

- **Split /19 and announce as two /20s, one on each link**

  basic inbound loadsharing

- **Example has no practical use, but demonstrates the principles**

Two links to the same ISP

- **Router A Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.240.0
 network 221.10.16.0 mask 255.255.240.0
 neighbor 222.222.10.2 remote-as 100
 neighbor 222.222.10.2 prefix-list routerC out
 neighbor 222.222.10.2 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerC permit 221.10.0.0/20
!
ip route 221.10.0.0 255.255.240.0 null0
ip route 221.10.16.0 255.255.240.0 null0
```

Two links to the same ISP

- **Router B Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.240.0
 network 221.10.16.0 mask 255.255.240.0
 neighbor 222.222.10.6 remote-as 100
 neighbor 222.222.10.6 prefix-list routerD out
 neighbor 222.222.10.6 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerD permit 221.10.16.0/20
!
ip route 221.10.0.0 255.255.240.0 null0
ip route 221.10.16.0 255.255.240.0 null0
```

Two links to the same ISP

- **Router C Configuration**

```
router bgp 100
 neighbor 222.222.10.1 remote-as 65534
 neighbor 222.222.10.1 default-originate
 neighbor 222.222.10.1 prefix-list Customer in
 neighbor 222.222.10.1 prefix-list default out
!
ip prefix-list Customer permit 221.10.0.0/20
ip prefix-list default permit 0.0.0.0/0
```

Two links to the same ISP

- **Router D Configuration**

```
router bgp 100
 neighbor 222.222.10.5 remote-as 65534
 neighbor 222.222.10.5 default-originate
 neighbor 222.222.10.5 prefix-list Customer in
 neighbor 222.222.10.5 prefix-list default out
!
ip prefix-list Customer permit 221.10.16.0/20
ip prefix-list default permit 0.0.0.0/0
```

Two links to the same ISP

- **Router E is AS100 border router**
  - **removes prefixes in the private AS from external announcements**
  - **implements the proxy aggregation for the customer prefixes**

Two links to the same ISP

- **Router E Configuration**

```
router bgp 100
 network 221.10.0.0 mask 255.255.224.0
 neighbor 222.222.10.17 remote-as 110
 neighbor 222.222.10.17 filter-list 1 out
!
ip route 221.10.0.0 255.255.224.0 null0
!
ip as-path access-list 1 deny ^65534$
ip as-path access-list 1 permit ^$
```

- **Private AS still visible inside AS100**

Two links to the same ISP

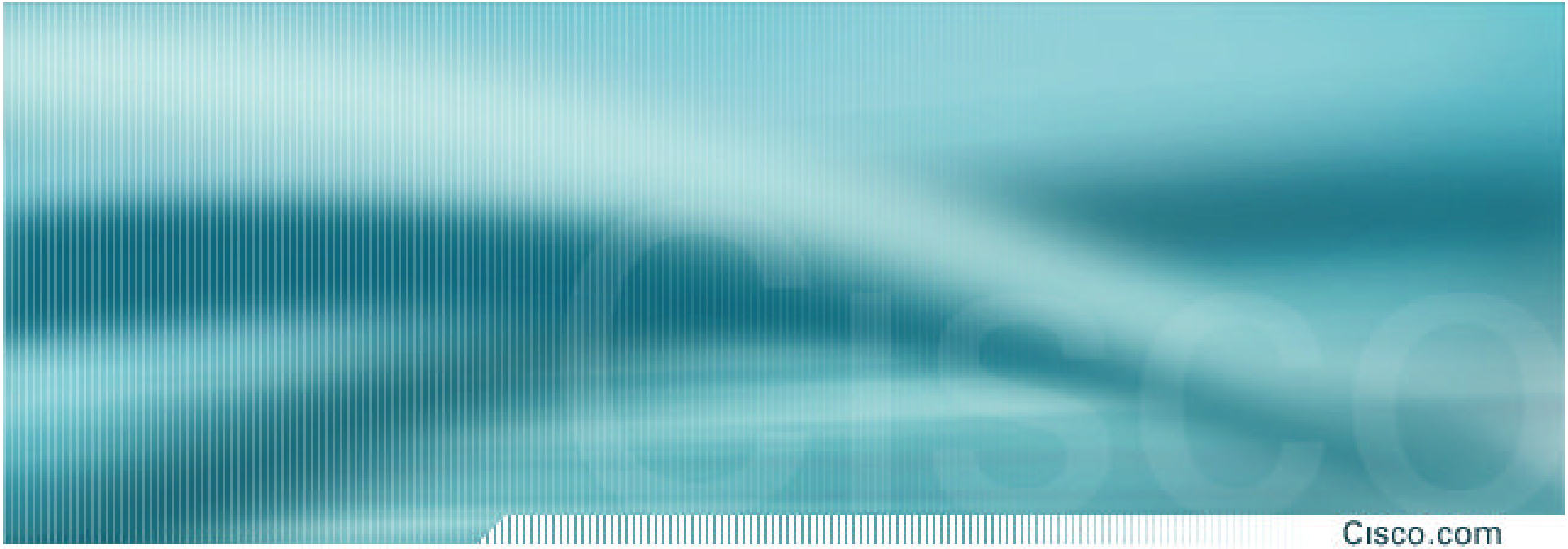
- **Big Problem:**

  no backup in case of link failure

- **/19 address block not announced**
- **AS Path filtering “awkward”**

  easier to use bgp command

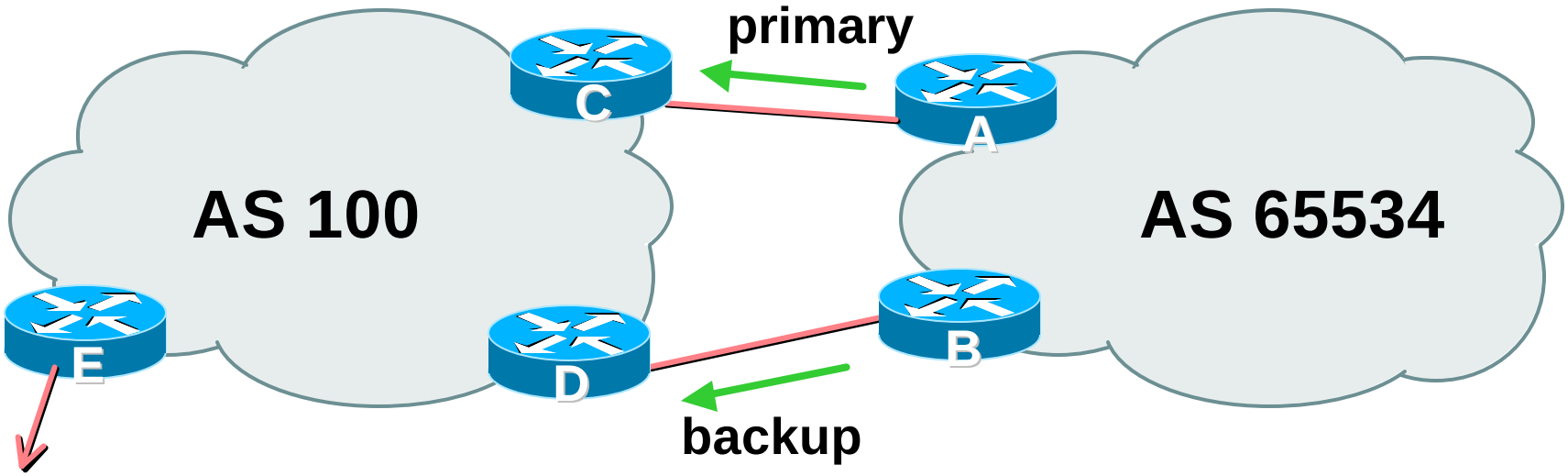
  ```
  neighbor x.x.x.x remove-private-AS
  ```



Two links to the same ISP

One link primary, the other link backup only

Two links to the same ISP



- **AS109 removes private AS and any customer subprefixes from Internet announcement**

Two links to the same ISP (one as backup only)

- **Announce /19 aggregate on each link**
  - **primary link makes standard announcement**
  - **backup link increases metric on outbound, and reduces local-pref on inbound**
- **When one link fails, the announcement of the /19 aggregate via the other link ensures continued connectivity**

Two links to the same ISP (one as backup only)

- **Router A Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 neighbor 222.222.10.2 remote-as 100
 neighbor 222.222.10.2 description RouterC
 neighbor 222.222.10.2 prefix-list aggregate out
 neighbor 222.222.10.2 prefix-list default in
!
ip prefix-list aggregate permit 221.10.0.0/19
ip prefix-list default permit 0.0.0.0/0
!
```

Two links to the same ISP (one as backup only)

- **Router B Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 neighbor 222.222.10.6 remote-as 100
 neighbor 222.222.10.6 description RouterD
 neighbor 222.222.10.6 prefix-list aggregate out
 neighbor 222.222.10.6 route-map routerD-out out
 neighbor 222.222.10.6 prefix-list default in
 neighbor 222.222.10.6 route-map routerD-in in
!
..next slide
```

Two links to the same ISP (one as backup only)


```
ip prefix-list aggregate permit 221.10.0.0/19
ip prefix-list default permit 0.0.0.0/0
!
route-map routerD-out permit 10
 match ip address prefix-list aggregate
 set metric 10
route-map routerD-out permit 20
!
route-map routerD-in permit 10
 set local-preference 90
!
```

Two links to the same ISP (one as backup only)

- **Router C Configuration (main link)**

```
router bgp 100
 neighbor 222.222.10.1 remote-as 65534
 neighbor 222.222.10.1 default-originate
 neighbor 222.222.10.1 prefix-list Customer in
 neighbor 222.222.10.1 prefix-list default out
!
ip prefix-list Customer permit 221.10.0.0/19
ip prefix-list default permit 0.0.0.0/0
```

Two links to the same ISP (one as backup only)

- **Router D Configuration (backup link)**

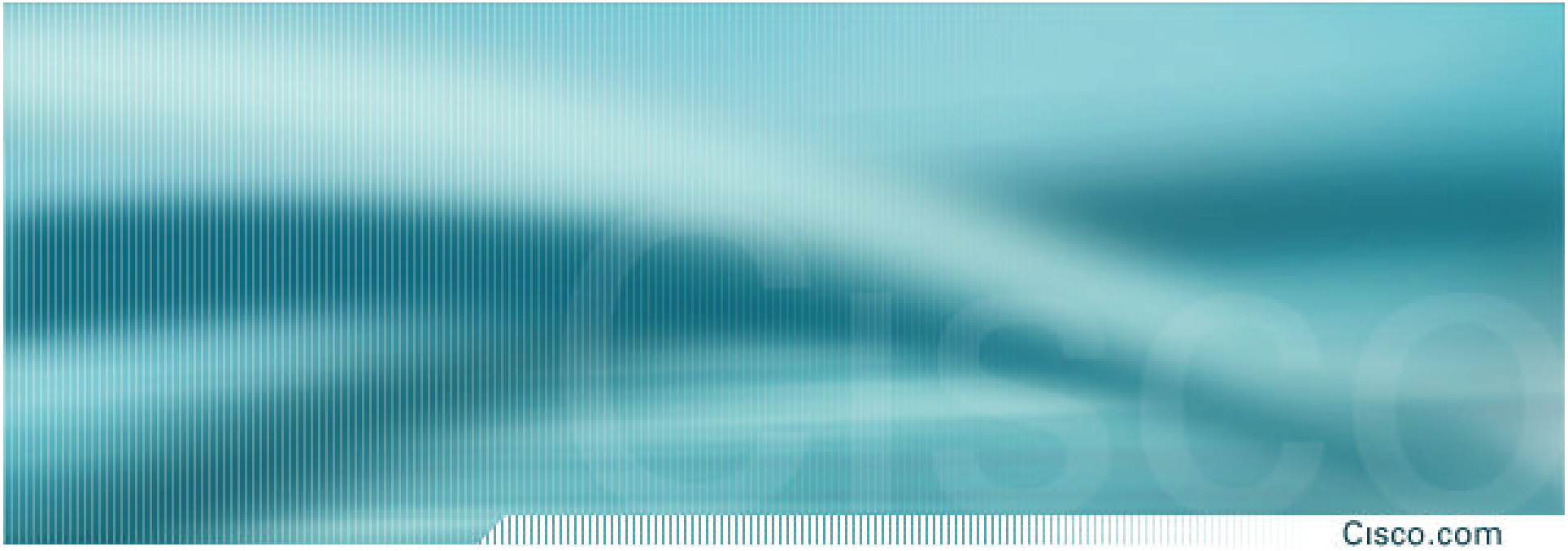
```
router bgp 100
 neighbor 222.222.10.5 remote-as 65534
 neighbor 222.222.10.5 default-originate
 neighbor 222.222.10.5 prefix-list Customer in
 neighbor 222.222.10.5 prefix-list default out
!
ip prefix-list Customer permit 221.10.0.0/19
ip prefix-list default permit 0.0.0.0/0
```

Two links to the same ISP (one as backup only)

- **Router E Configuration**

```
router bgp 100
 neighbor 222.222.10.17 remote-as 110
 neighbor 222.222.10.17 remove-private-AS
 neighbor 222.222.10.17 prefix-list Customer out
!
ip prefix-list Customer permit 221.10.0.0/19
```

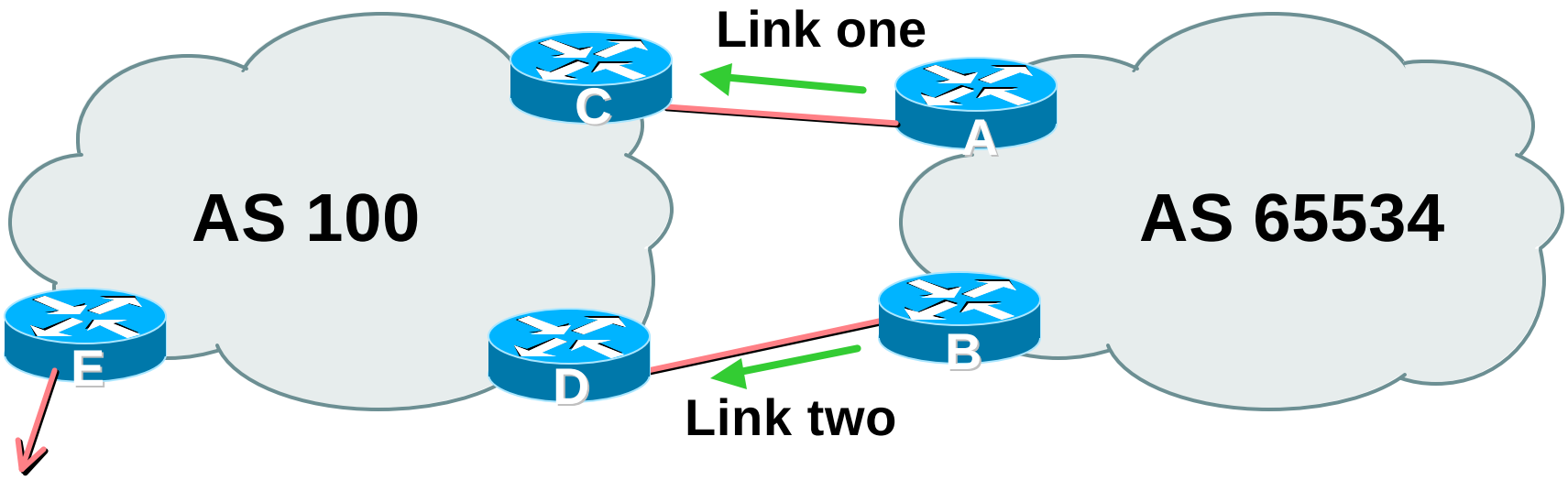
- **Router E removes the private AS and customer's subprefixes from external announcements**
- **Private AS still visible inside AS100**



Two links to the same ISP

With Redundancy and Loadsharing

Two links to the same ISP (with redundancy)



- **AS100 removes private AS and any customer subprefixes from Internet announcement**

Loadsharing to the same ISP

- Announce /19 aggregate on each link
- Split /19 and announce as two /20s, one on each link
  - basic inbound loadsharing
  - assumes equal circuit capacity and even spread of traffic across address block
- Vary the split until “perfect” loadsharing achieved
- Accept the default from upstream
  - basic outbound loadsharing by nearest exit
  - okay in first approx as most ISP and end-site traffic is inbound

Two links to the same ISP

- **Router A Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.0.0 mask 255.255.240.0
 neighbor 222.222.10.2 remote-as 100
 neighbor 222.222.10.2 prefix-list routerC out
 neighbor 222.222.10.2 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerC permit 221.10.0.0/20
ip prefix-list routerC permit 221.10.0.0/19
!
ip route 221.10.0.0 255.255.240.0 null0
ip route 221.10.0.0 255.255.224.0 null0
```

Two links to the same ISP

- **Router B Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.16.0 mask 255.255.240.0
 neighbor 222.222.10.6 remote-as 100
 neighbor 222.222.10.6 prefix-list routerD out
 neighbor 222.222.10.6 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerD permit 221.10.16.0/20
ip prefix-list routerD permit 221.10.0.0/19
!
ip route 221.10.0.0 255.255.224.0 null0
ip route 221.10.16.0 255.255.240.0 null0
```

Loadsharing to the same ISP

- **Default route for outbound traffic?**

  Use default-information originate for the IGP and rely on IGP metrics for nearest exit

  e.g. on router A:

```
router ospf 65534
 default-information originate metric 2 metric-type 1
```

Two links to the same ISP

- **Router C Configuration**

```
router bgp 100
 neighbor 222.222.10.1 remote-as 65534
 neighbor 222.222.10.1 default-originate
 neighbor 222.222.10.1 prefix-list Customer in
 neighbor 222.222.10.1 prefix-list default out
!
ip prefix-list Customer permit 221.10.0.0/19 le 20
ip prefix-list default permit 0.0.0.0/0
```

- **Router C only allows in /19 and /20 prefixes from customer block**

Two links to the same ISP

- **Router D Configuration**

```
router bgp 100
 neighbor 222.222.10.5 remote-as 65534
 neighbor 222.222.10.5 default-originate
 neighbor 222.222.10.5 prefix-list Customer in
 neighbor 222.222.10.5 prefix-list default out
!
ip prefix-list Customer permit 221.10.0.0/19 le 20
ip prefix-list default permit 0.0.0.0/0
```

- **Router D only allows in /19 and /20 prefixes from customer block**

Two links to the same ISP

- **Router E is AS100 border router**
  - **removes subprefixes in the private AS from external announcements**
  - **removes the private AS from external announcement of the customer /19**

Two links to the same ISP (with redundancy)

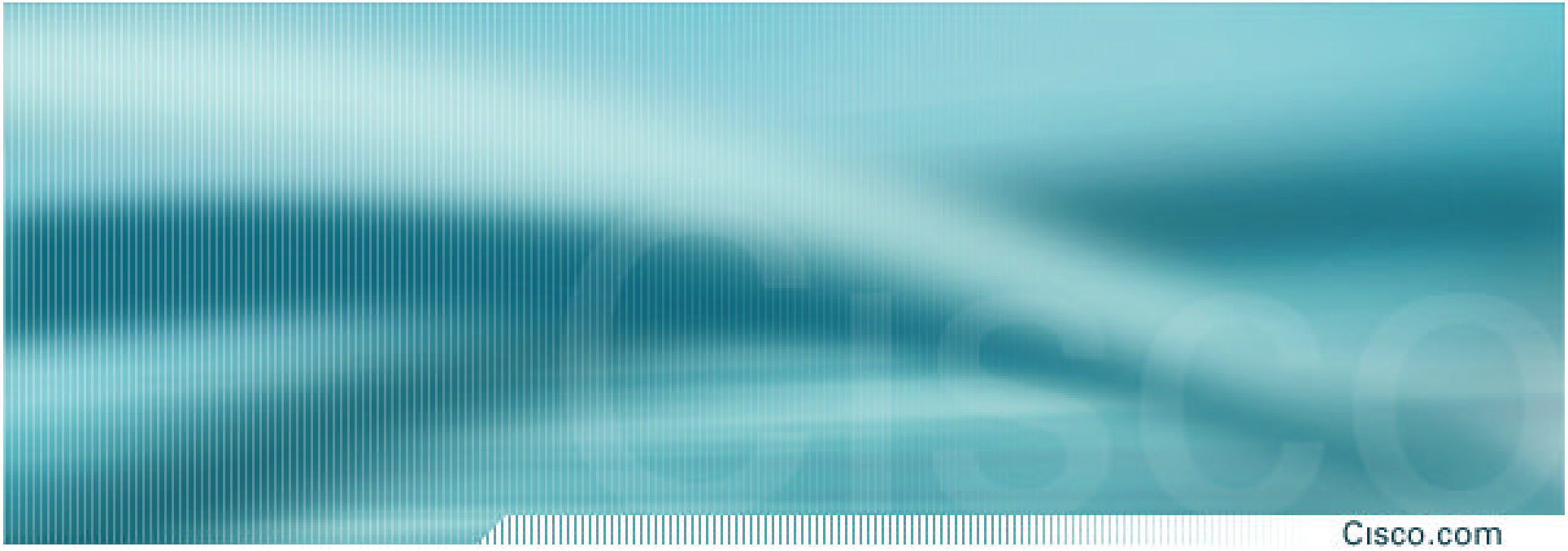
- **Router E Configuration**

```
router bgp 100
 neighbor 222.222.10.17 remote-as 110
 neighbor 222.222.10.17 remove-private-AS
 neighbor 222.222.10.17 prefix-list Customer out
!
ip prefix-list Customer permit 221.10.0.0/19
```

- **Private AS still visible inside AS100**

Loadsharing to the same ISP

- **Loadsharing configuration is only on customer router**
- **Upstream ISP has to**
  - **remove customer subprefixes from external announcements**
  - **remove private AS from external announcements**
- **Could also use BGP communities**

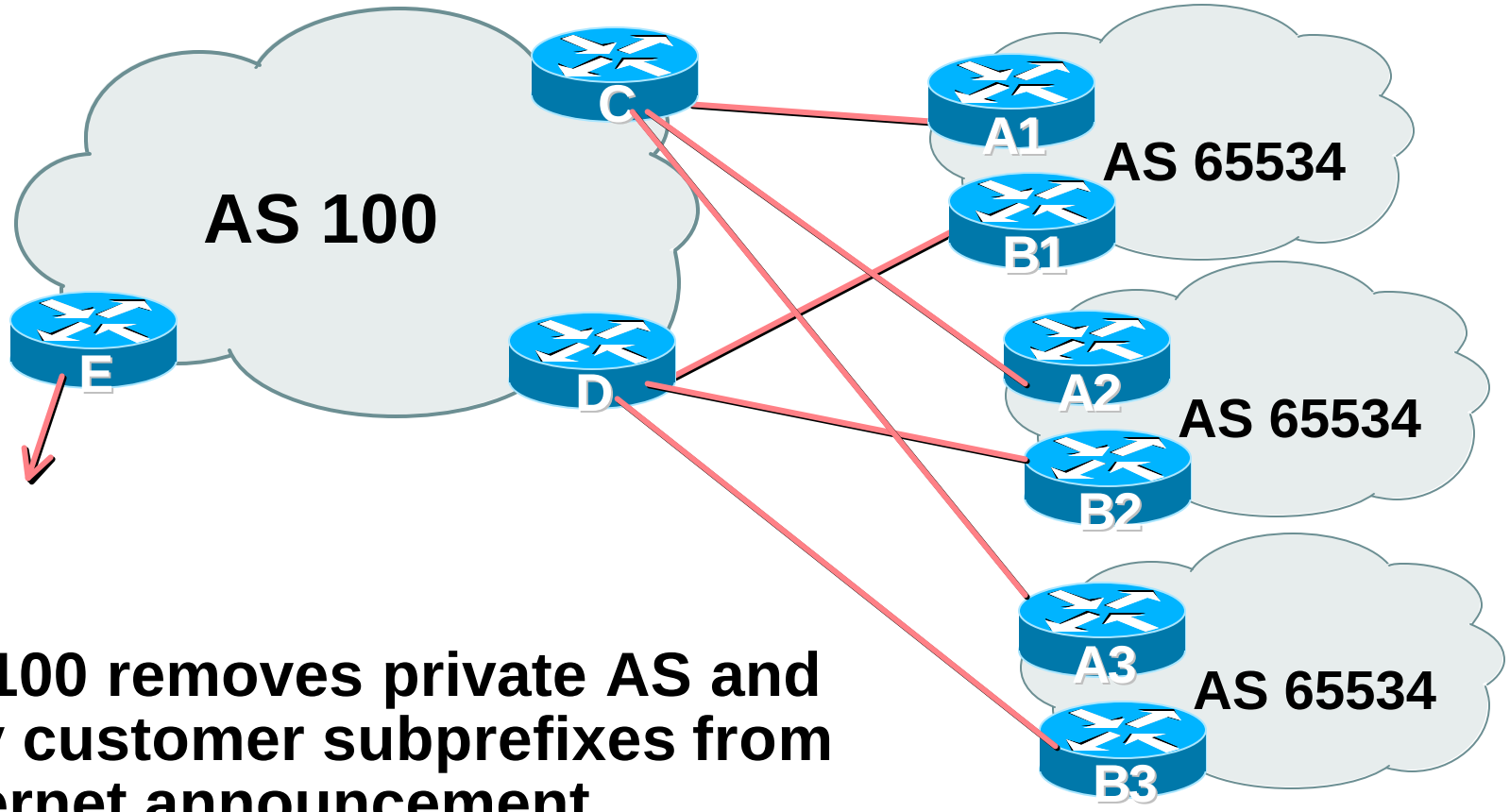


Two links to the same ISP

Multiple Dualhomed Customers

(RFC2270)

Multiple Dualhomed Customers (RFC2270)



- **AS100 removes private AS and any customer subprefixes from Internet announcement**

Multiple Dualhomed Customers

- **Customer announcements as per previous example**
- **Use the *same* private AS for each customer**
  - **documented in RFC2270**
  - **address space is not overlapping**
  - **each customer hears default only**
- **Router A*n* and B*n* configuration same as Router A and B previously**

Two links to the same ISP

- **Router A1 Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.0.0 mask 255.255.240.0
 neighbor 222.222.10.2 remote-as 100
 neighbor 222.222.10.2 prefix-list routerC out
 neighbor 222.222.10.2 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerC permit 221.10.0.0/20
ip prefix-list routerC permit 221.10.0.0/19
!
ip route 221.10.0.0 255.255.240.0 null0
ip route 221.10.0.0 255.255.224.0 null0
```

Two links to the same ISP

- **Router B1 Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.16.0 mask 255.255.240.0
 neighbor 222.222.10.6 remote-as 100
 neighbor 222.222.10.6 prefix-list routerD out
 neighbor 222.222.10.6 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerD permit 221.10.16.0/20
ip prefix-list routerD permit 221.10.0.0/19
!
ip route 221.10.0.0 255.255.224.0 null0
ip route 221.10.16.0 255.255.240.0 null0
```

Multiple Dualhomed Customers

Cisco.com

- **Router C Configuration**

```
router bgp 100
 neighbor bgp-customers peer-group
 neighbor bgp-customers remote-as 65534
 neighbor bgp-customers default-originate
 neighbor bgp-customers prefix-list default out
 neighbor 222.222.10.1 peer-group bgp-customers
 neighbor 222.222.10.1 description Customer One
 neighbor 222.222.10.1 prefix-list Customer1 in
 neighbor 222.222.10.9 peer-group bgp-customers
 neighbor 222.222.10.9 description Customer Two
 neighbor 222.222.10.9 prefix-list Customer2 in
```

Multiple Dualhomed Customers


```
 neighbor 222.222.10.17 peer-group bgp-customers
 neighbor 222.222.10.17 description Customer Three
 neighbor 222.222.10.17 prefix-list Customer3 in
!
ip prefix-list Customer1 permit 221.10.0.0/19 le 20
ip prefix-list Customer2 permit 221.16.64.0/19 le 20
ip prefix-list Customer3 permit 221.14.192.0/19 le 20
ip prefix-list default permit 0.0.0.0/0
```

- Router C only allows in /19 and /20 prefixes from customer block

Multiple Dualhomed Customers

Cisco.com

- **Router D Configuration**

```
router bgp 100
 neighbor bgp-customers peer-group
 neighbor bgp-customers remote-as 65534
 neighbor bgp-customers default-originate
 neighbor bgp-customers prefix-list default out
 neighbor 222.222.10.5 peer-group bgp-customers
 neighbor 222.222.10.5 description Customer One
 neighbor 222.222.10.5 prefix-list Customer1 in
 neighbor 222.222.10.13 peer-group bgp-customers
 neighbor 222.222.10.13 description Customer Two
 neighbor 222.222.10.13 prefix-list Customer2 in
```

Multiple Dualhomed Customers

Cisco.com

```
 neighbor 222.222.10.21 peer-group bgp-customers
 neighbor 222.222.10.21 description Customer Three
 neighbor 222.222.10.21 prefix-list Customer3 in
!
ip prefix-list Customer1 permit 221.10.0.0/19 le 20
ip prefix-list Customer2 permit 221.16.64.0/19 le 20
ip prefix-list Customer3 permit 221.14.192.0/19 le 20
ip prefix-list default permit 0.0.0.0/0
```

- **Router D only allows in /19 and /20 prefixes from customer block**

Multiple Dualhomed Customers

- **Router E Configuration**

  assumes customer address space is not part of upstream's address block

```
router bgp 100
 neighbor 222.222.10.17 remote-as 110
 neighbor 222.222.10.17 remove-private-AS
 neighbor 222.222.10.17 prefix-list Customers out
!
ip prefix-list Customers permit 221.10.0.0/19
ip prefix-list Customers permit 221.16.64.0/19
ip prefix-list Customers permit 221.14.192.0/19
```

- **Private AS still visible inside AS100**

Multiple Dualhomed Customers

- **If customers' prefixes come from ISP's address block**

  do NOT announce them to the Internet

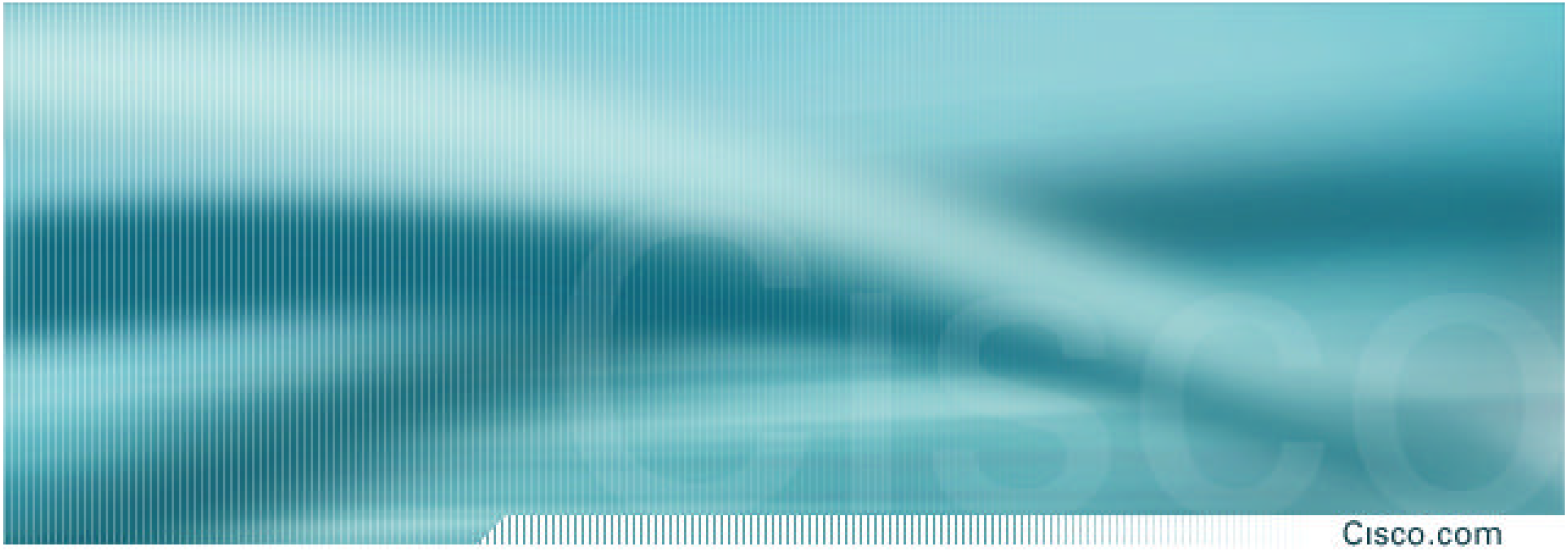
  announce ISP aggregate only

- **Router E configuration:**

```
router bgp 100
 neighbor 222.222.10.17 remote-as 110
 neighbor 222.222.10.17 prefix-list my-aggregate out
!
ip prefix-list my-aggregate permit 221.8.0.0/13
```

Multihoming Summary

- Use private AS for multihoming to upstream
- Leak subprefixes to upstream only to aid loadsharing
- Upstream Router E configuration is uniform across all scenarios



Two links to different ISPs

Two links to different ISPs

- **Use Public ASes**

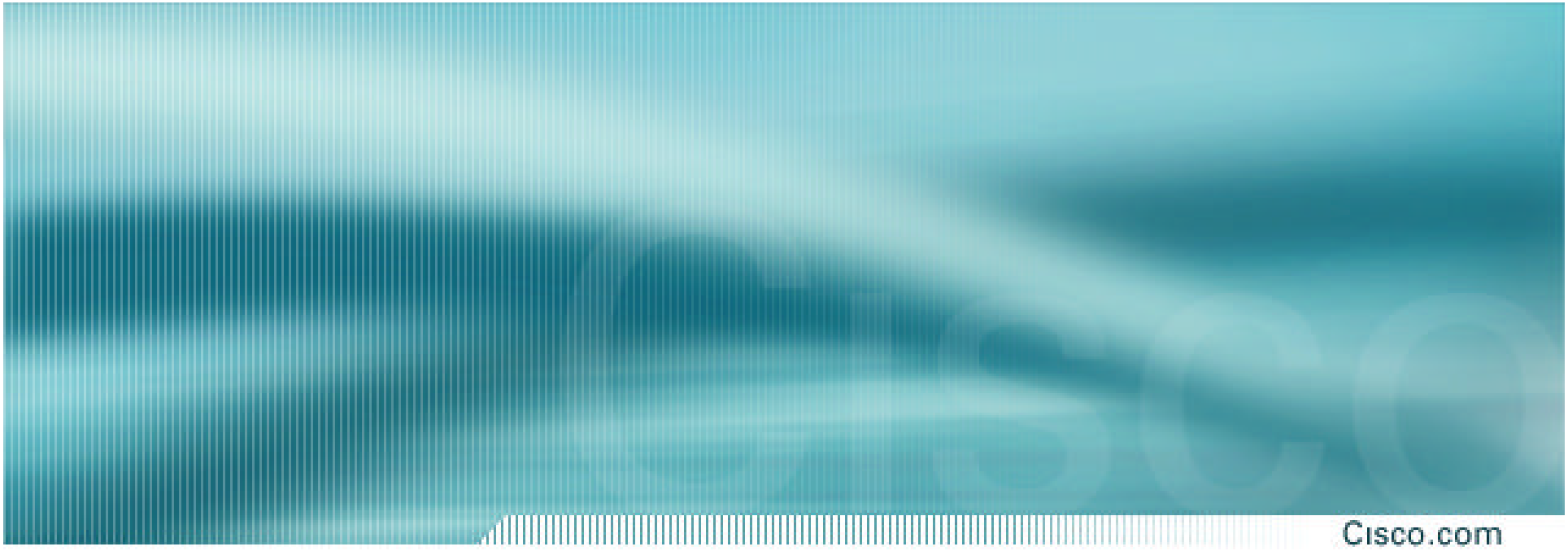
  or use private AS if agreed with the other ISP

- **Address space comes from**

  both upstreams or

  Regional Internet Registry

- **Configuration concepts very similar**



Two links to different ISPs

Basic – No Redundancy

Two links to different ISPs

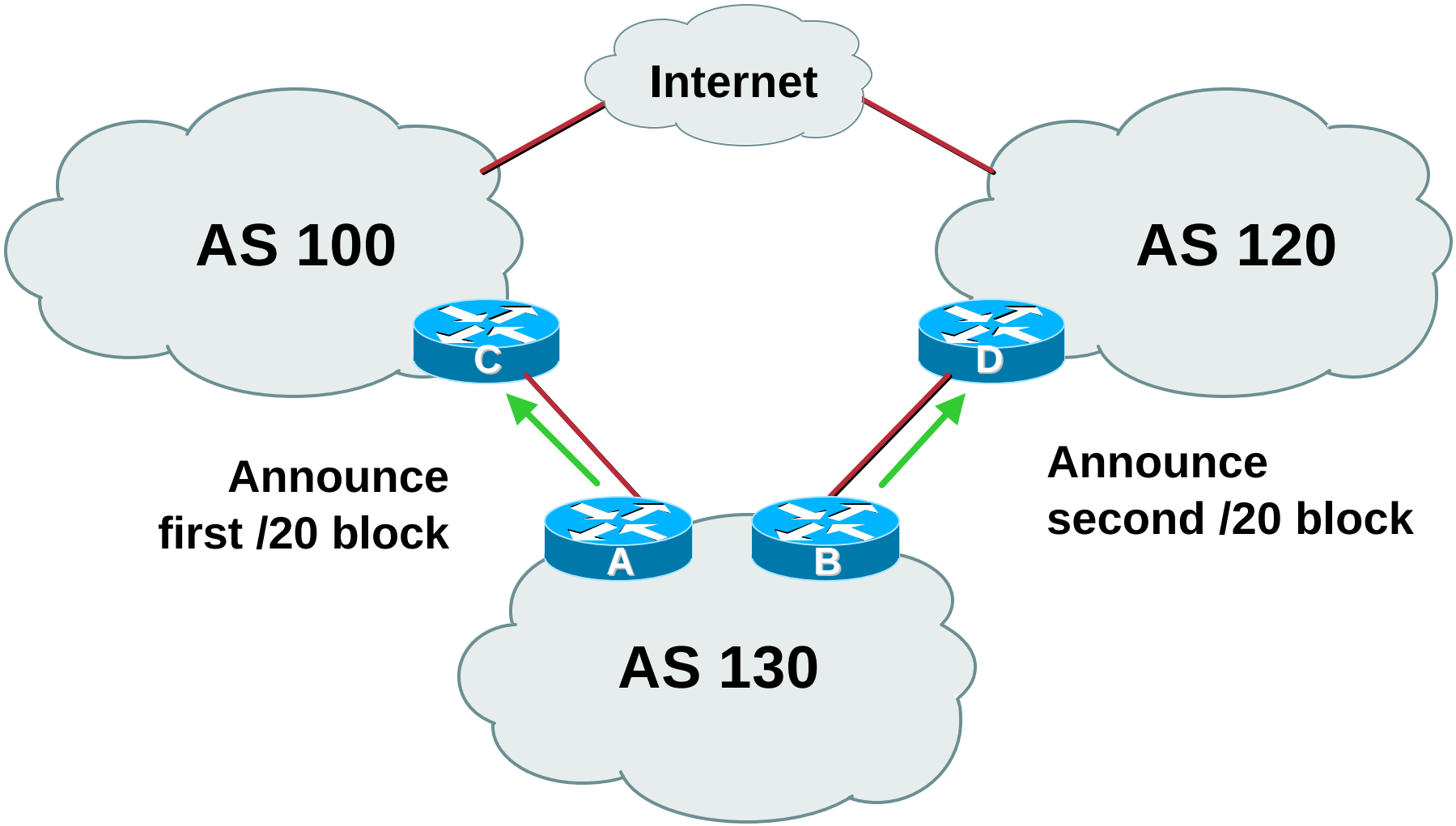
- **Example for PI space**

  ISP network, or large enterprise site

- **Split /19 and announce as two /20s, one on each link**

  basic inbound loadsharing

Two links to different ISPs



Two links to different ISPs

- **Router A Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.240.0
 neighbor 222.222.10.1 remote-as 100
 neighbor 222.222.10.1 prefix-list routerC out
 neighbor 222.222.10.1 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerC permit 221.10.0.0/20
```

Two links to different ISPs

- **Router B Configuration**

```
router bgp 130
 network 221.10.16.0 mask 255.255.240.0
 neighbor 220.1.5.1 remote-as 120
 neighbor 220.1.5.1 prefix-list routerD out
 neighbor 220.1.5.1 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
ip prefix-list routerD permit 221.10.16.0/20
```

Two links to different ISPs

- **Router C Configuration**

```
router bgp 100
 neighbor 221.10.1.1 remote-as 130
 neighbor 221.10.1.1 default-originate
 neighbor 221.10.1.1 prefix-list AS130cust in
 neighbor 221.10.1.1 prefix-list default-out out
!
```

- **Router C only announces default to AS 130**
- **Router C only accepts AS130's prefix block**

Two links to different ISPs

- **Router D Configuration**

```
router bgp 120
 neighbor 220.1.5.1 remote-as 130
 neighbor 220.1.5.1 default-originate
 neighbor 220.1.5.1 prefix-list AS130cust in
 neighbor 220.1.5.1 prefix-list default-out out
!
```

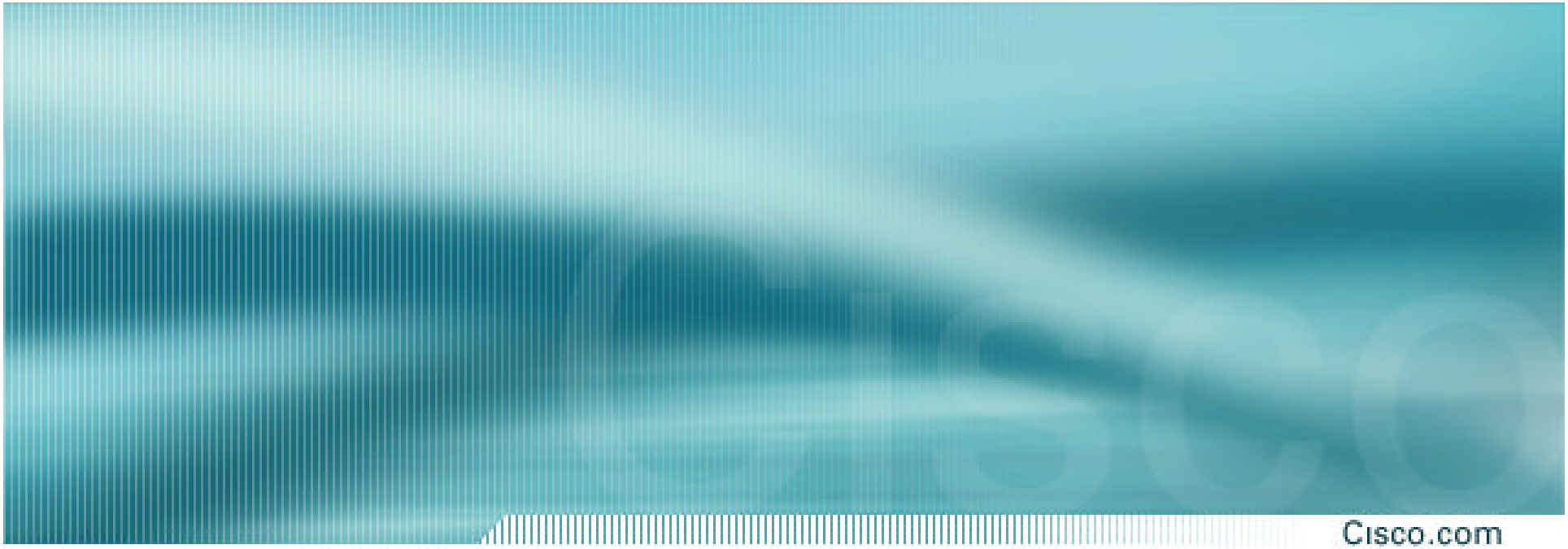
- **Router D only announces default to AS 130**
- **Router D only accepts AS130's prefix block**

Two links to different ISPs (no redundancy)

- **Big Problem:**

  no backup in case of link failure

- **/19 address block not announced**



Two links to different ISPs

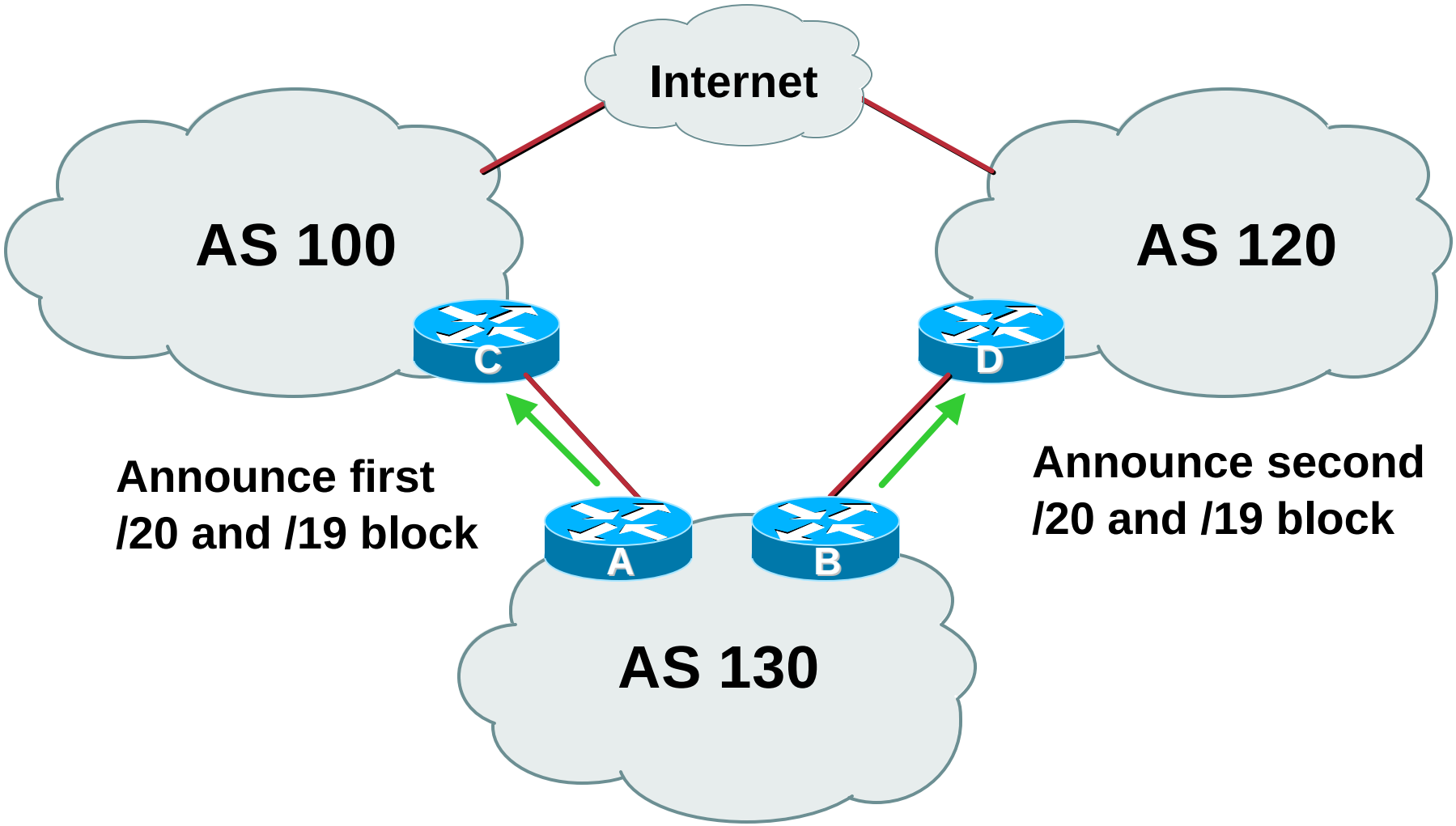
With Redundancy

Two links to different ISPs (with redundancy)

- Announce /19 aggregate on each link
- Split /19 and announce as two /20s, one on each link

  basic inbound loadsharing
- When one link fails, the announcement of the /19 aggregate via the other ISP ensures continued connectivity

Two links to different ISPs (with redundancy)



Two links to different ISPs (with redundancy)

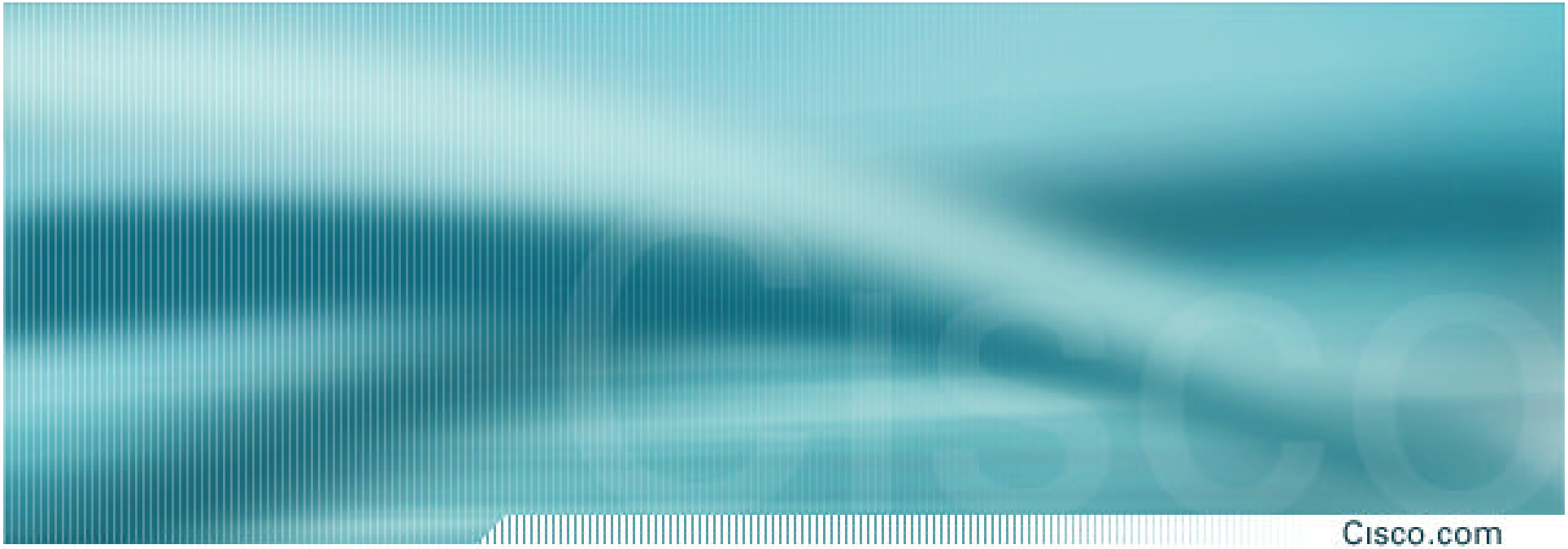
- **Router A Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.0.0 mask 255.255.240.0
 neighbor 222.222.10.1 remote-as 100
 neighbor 222.222.10.1 prefix-list firstblock out
 neighbor 222.222.10.1 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
!
ip prefix-list firstblock permit 221.10.0.0/20
ip prefix-list firstblock permit 221.10.0.0/19
```

Two links to different ISPs (with redundancy)

- **Router B Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.16.0 mask 255.255.240.0
 neighbor 220.1.5.1 remote-as 120
 neighbor 220.1.5.1 prefix-list secondblock out
 neighbor 220.1.5.1 prefix-list default in
!
ip prefix-list default permit 0.0.0.0/0
!
ip prefix-list secondblock permit 221.10.16.0/20
ip prefix-list secondblock permit 221.10.0.0/19
```



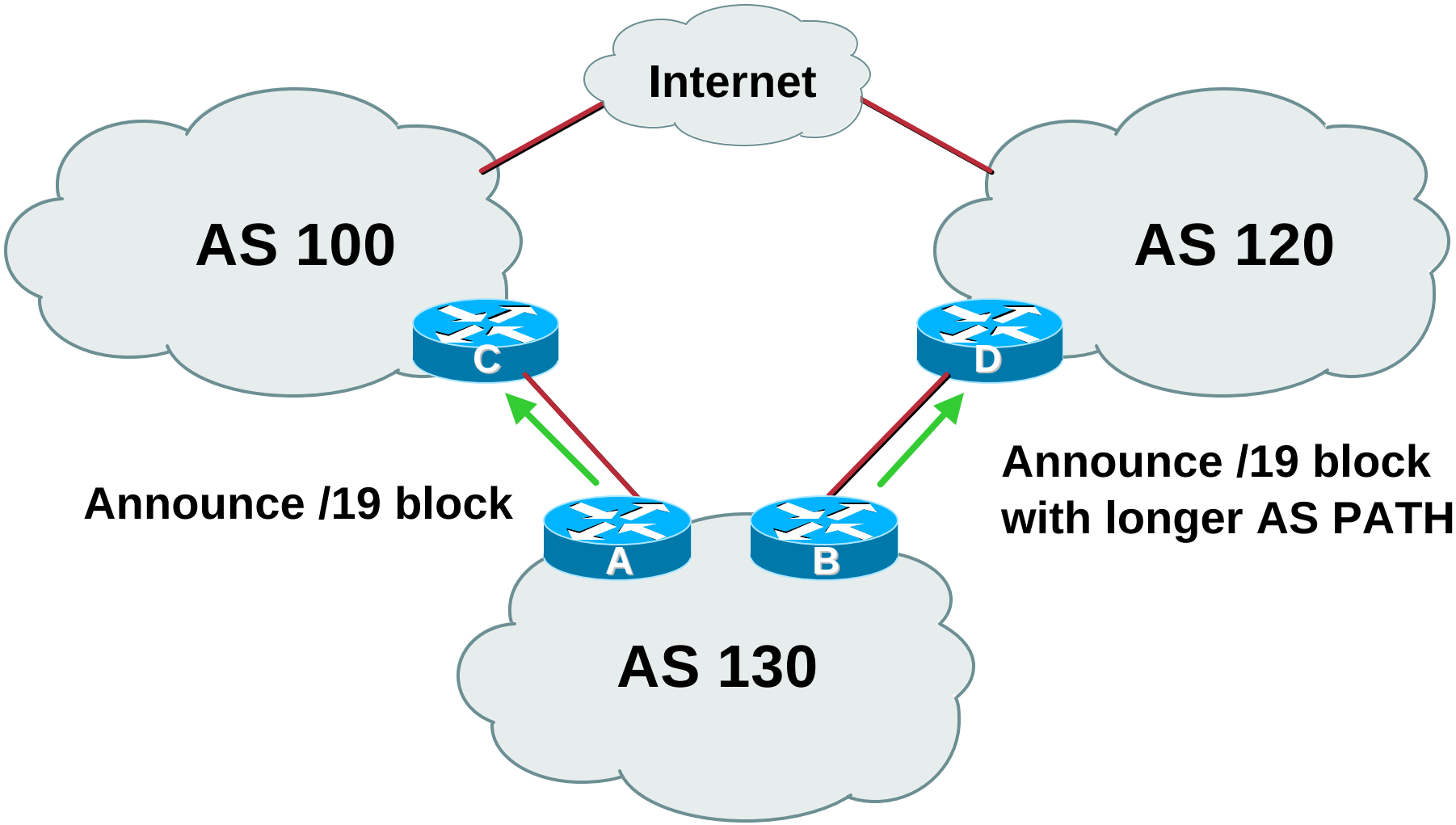
Two links to different ISPs

One link primary, the other link backup only

Two links to different ISPs (one as backup only)

- Announce /19 aggregate on each link
  - primary link makes standard announcement
  - backup link lengthens the AS PATH by using AS PATH prepend
- When one link fails, the announcement of the /19 aggregate via the other link ensures continued connectivity

Two links to different ISPs (one as backup only)



Two links to different ISPs (one as backup only)

- **Router A Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 neighbor 222.222.10.1 remote-as 100
 neighbor 222.222.10.1 prefix-list aggregate out
 neighbor 222.222.10.1 prefix-list default in
!
ip prefix-list aggregate permit 221.10.0.0/19
ip prefix-list default permit 0.0.0.0/0
```

Two links to different ISPs (one as backup only)

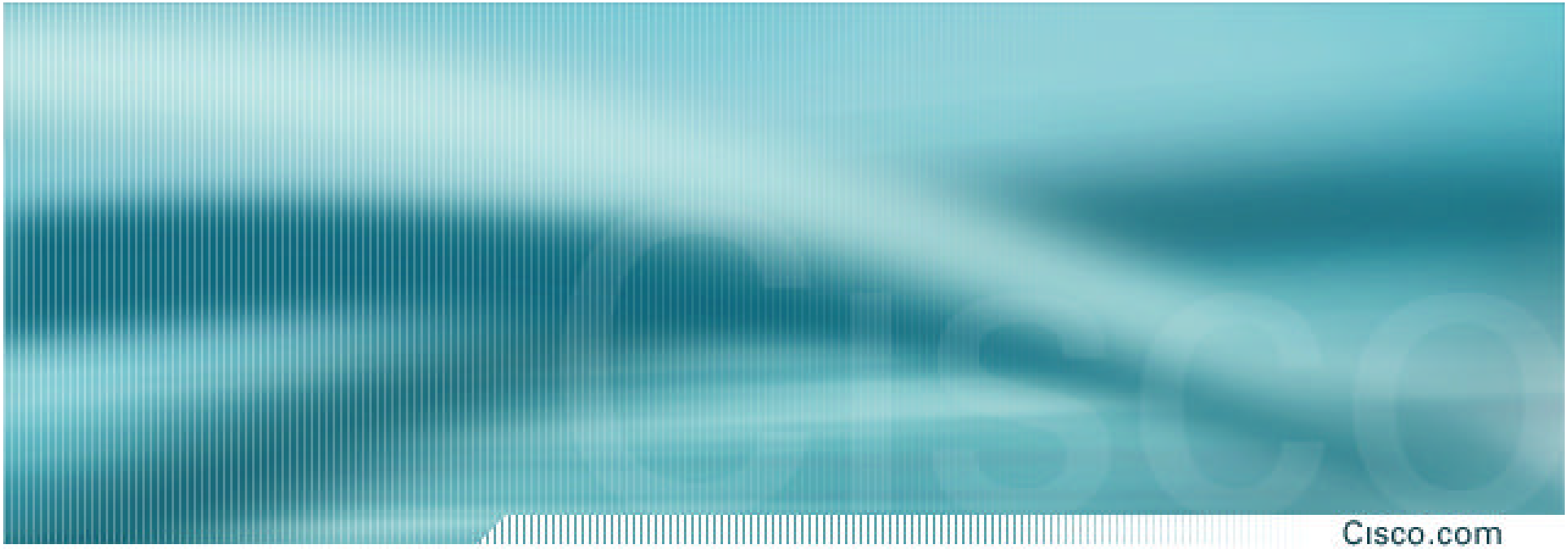
- **Router B Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 neighbor 220.1.5.1 remote-as 120
 neighbor 220.1.5.1 prefix-list aggregate out
 neighbor 220.1.5.1 route-map routerD-out out
 neighbor 220.1.5.1 prefix-list default in
 neighbor 220.1.5.1 route-map routerD-in in
!
..next slide
```

Two links to different ISPs (one as backup only)

- **Router B Configuration**

```
!
ip prefix-list aggregate permit 221.10.0.0/19
ip prefix-list default permit 0.0.0.0/0
!
route-map routerD-out permit 10
 set as-path prepend 130 130 130
!
route-map routerD-in permit 10
 set local-preference 80
```



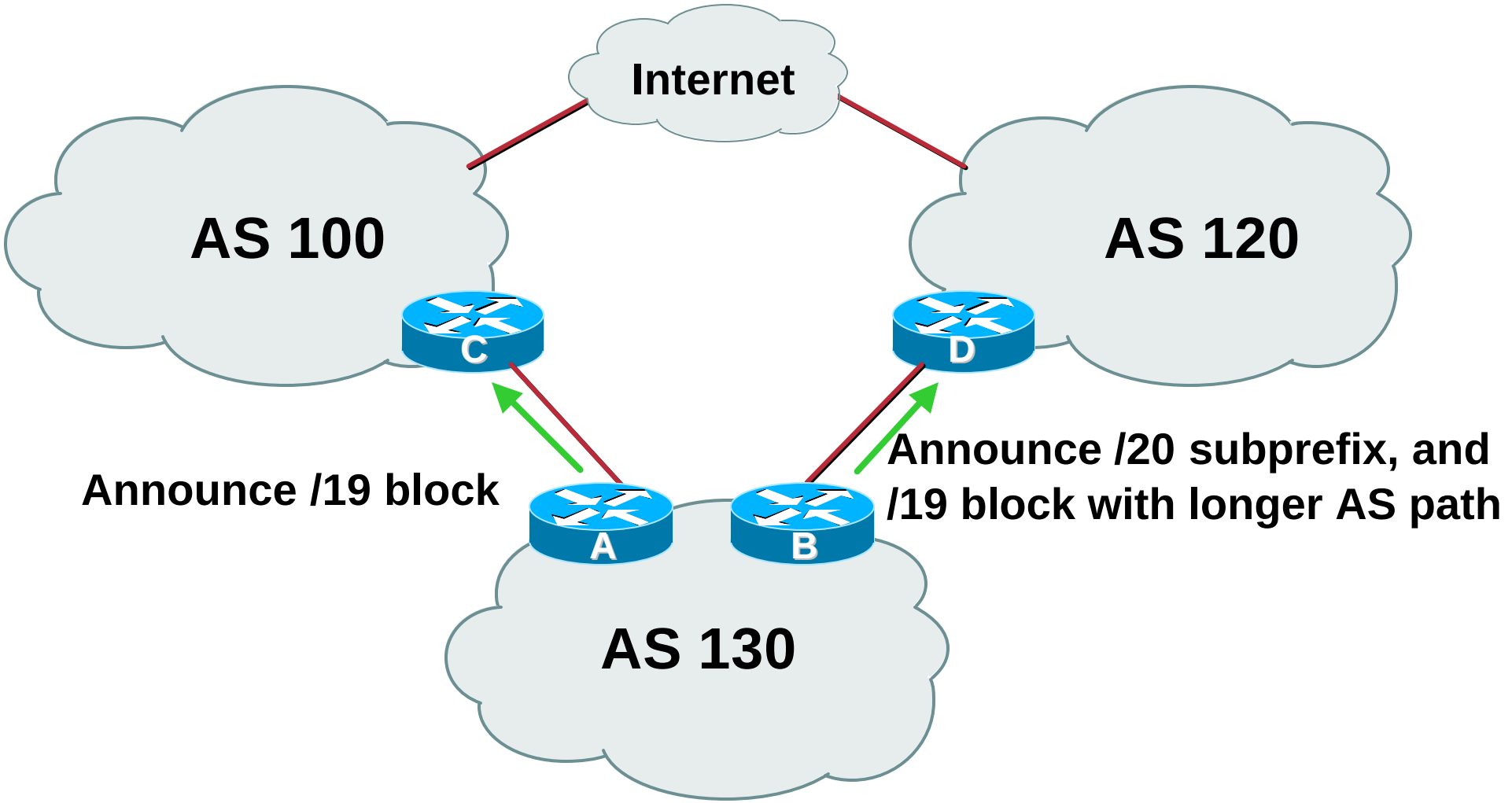
Two links to different ISPs

More Controlled Loadsharing

Loadsharing with different ISPs

- **Announce /19 aggregate on each link**
  - **On first link, announce /19 as normal**
  - **On second link, announce /19 with longer AS PATH, and announce one /20 subprefix**
    - **controls loadsharing between upstreams and the Internet**
- **Vary the subprefix size and AS PATH length until “perfect” loadsharing achieved**
- **Still require redundancy!**

Loadsharing with different ISPs



Loadsharing with different ISPs

- **Router A Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 neighbor 222.222.10.1 remote-as 100
 neighbor 222.222.10.1 prefix-list default in
 neighbor 222.222.10.1 prefix-list aggregate out
!
ip prefix-list aggregate permit 221.10.0.0/19
```

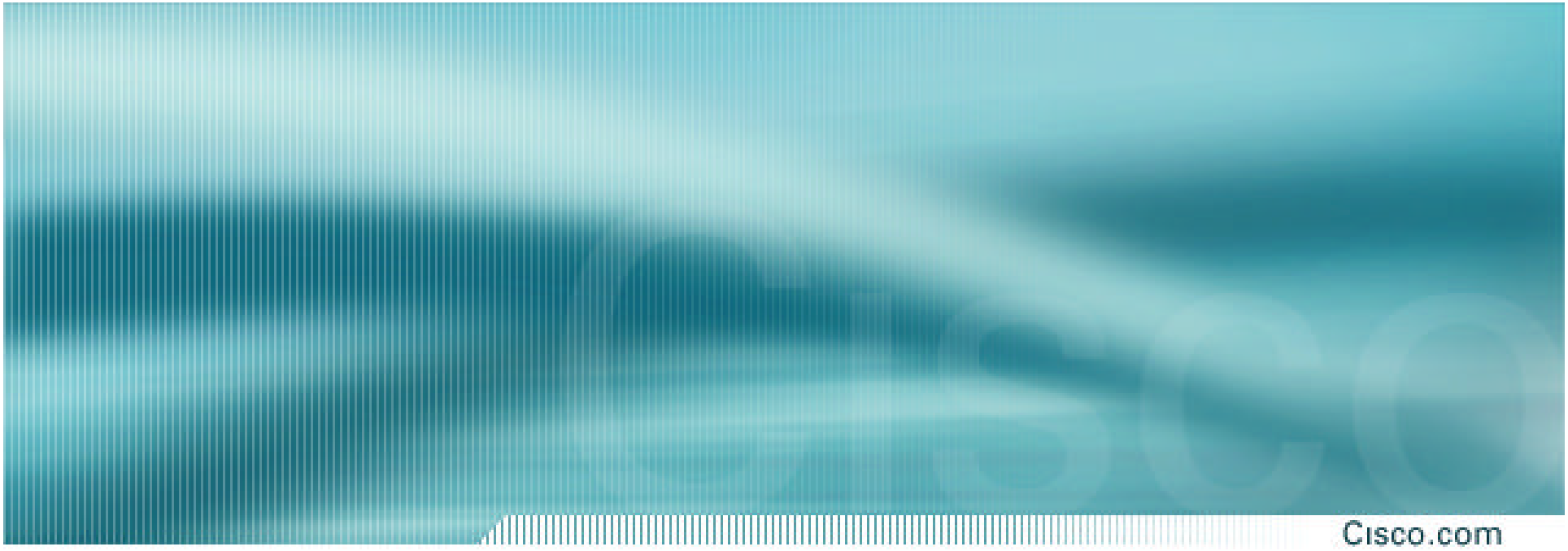
Loadsharing with different ISPs

- **Router B Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.16.0 mask 255.255.240.0
 neighbor 220.1.5.1 remote-as 120
 neighbor 220.1.5.1 prefix-list default in
 neighbor 220.1.5.1 prefix-list subblocks out
 neighbor 220.1.5.1 route-map routerD out
!
..next slide..
```

Loadsharing with different ISPs

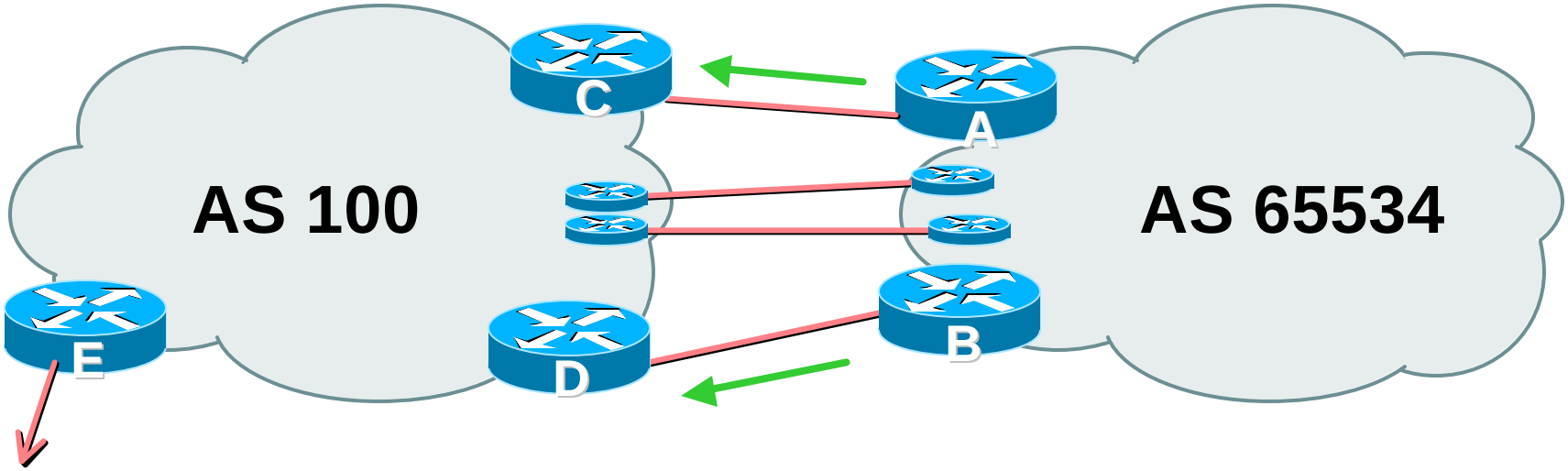

```
route-map routerD permit 10
 match ip address prefix-list aggregate
 set as-path prepend 130 130
route-map routerD permit 20
!
ip prefix-list subblocks permit 221.10.0.0/19 le 20
ip prefix-list aggregate permit 221.10.0.0/19
```



Loadsharing Using Communities

4 links – Private AS

Private AS



- **AS100 removes private AS and any customer subprefixes from Internet announcement**

Private AS

- Announce /19 aggregate on each link
- Split /19 and announce as four /21s, one on each link
  - basic inbound loadsharing
  - assumes equal circuit capacity and even spread of traffic across address block
- Vary the split until “perfect” loadsharing achieved
  - use the no-export community for subprefixes

Private AS

- **Router A Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.0.0 mask 255.255.248.0
 neighbor 222.222.10.2 remote-as 100
 neighbor 222.222.10.2 send-community
 neighbor 222.222.10.2 prefix-list subblocks1 out
 neighbor 222.222.10.2 route-map routerC-out out
 neighbor 222.222.10.2 prefix-list default in
!
..next slide
```

Private AS


```
ip prefix-list subblocks1 permit 221.10.0.0/19
ip prefix-list subblocks1 permit 221.10.0.0/21
!
ip prefix-list firstblock permit 221.10.0.0/21
ip prefix-list default permit 0.0.0.0/0
!
route-map routerC-out permit 10
 match ip address prefix-list firstblock
 set community no-export
route-map routerC-out permit 20
```

Private AS

Cisco.com

- **Router B Configuration**

```
router bgp 65534
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.24.0 mask 255.255.248.0
 neighbor 222.222.20.2 remote-as 100
 neighbor 222.222.20.2 send-community
 neighbor 222.222.20.2 prefix-list subblocks2 out
 neighbor 222.222.20.2 route-map routerD-out out
 neighbor 222.222.20.2 prefix-list default in
!
..next slide
```

Private AS


```
ip prefix-list subblocks2 permit 221.10.0.0/19
ip prefix-list subblocks2 permit 221.10.24.0/21
!
ip prefix-list secondblock permit 221.10.24.0/21
ip prefix-list default permit 0.0.0.0/0
!
route-map routerD-out permit 10
 match ip address prefix-list secondblock
 set community no-export
route-map routerD-out permit 20
```

Private AS

- **Router E Configuration**

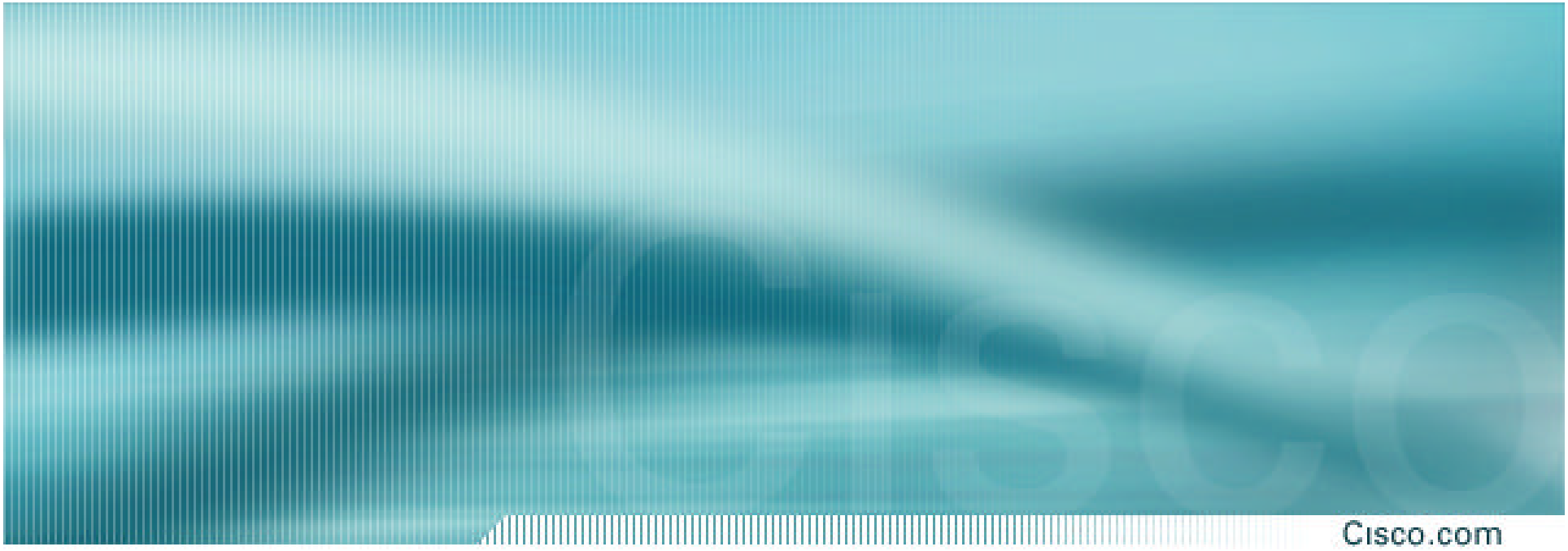
```
router bgp 100
 neighbor 222.222.10.17 remote-as 110
 neigbhor 222.222.10.17 remove-private-AS
!
```

- **Router E removes the private AS from external announcements**
- **Router E automatically removes subprefixes with no-export community set**
- **Private AS still visible inside AS100**

Private AS

Cisco.com

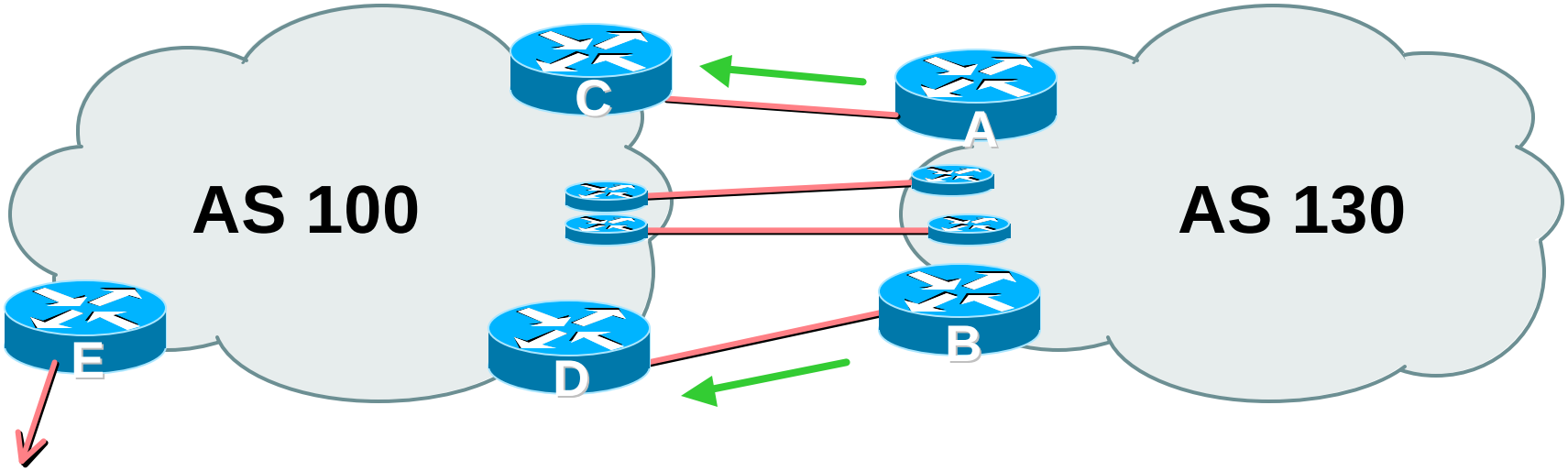
- Router C and D configuration is as previously
- AS100 routers will not advertise prefixes marked with community no-export to other ASes
- AS100 routers still need to filter the private AS
- Only a single /19 prefix is announced to the Internet - no routing table bloat! :-)



Loadsharing Using Communities

4 links – Public AS

Public AS



- 4 links between AS130 and AS100

Public AS

- Announce /19 aggregate on each link
- Split /19 and announce as four /21s, one on each link
  - basic inbound loadsharing
  - assumes equal circuit capacity and even spread of traffic across address block
- Vary the split until “perfect” loadsharing achieved
  - use the no-export community for subprefixes

Public AS

- **Router A Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.0.0 mask 255.255.248.0
 neighbor 222.222.10.2 remote-as 100
 neighbor 222.222.10.2 send-community
 neighbor 222.222.10.2 prefix-list subblocks1 out
 neighbor 222.222.10.2 route-map routerC-out out
 neighbor 222.222.10.2 prefix-list default in
!
..next slide
```

Public AS


```
ip prefix-list subblocks1 permit 221.10.0.0/19
ip prefix-list subblocks1 permit 221.10.0.0/21
!
ip prefix-list firstblock permit 221.10.0.0/21
ip prefix-list default permit 0.0.0.0/0
!
route-map routerC-out permit 10
 match ip address prefix-list firstblock
 set community no-export
route-map routerC-out permit 20
```

Public AS

- **Router B Configuration**

```
router bgp 130
 network 221.10.0.0 mask 255.255.224.0
 network 221.10.24.0 mask 255.255.248.0
 neighbor 222.222.20.2 remote-as 100
 neighbor 222.222.20.2 send-community
 neighbor 222.222.20.2 prefix-list subblocks2 out
 neighbor 222.222.20.2 route-map routerD-out out
 neighbor 222.222.20.2 prefix-list default in
!
..next slide
```

Public AS


```
ip prefix-list subblocks2 permit 221.10.0.0/19
ip prefix-list subblocks2 permit 221.10.24.0/21
!
ip prefix-list secondblock permit 221.10.24.0/21
ip prefix-list default permit 0.0.0.0/0
!
route-map routerD-out permit 10
 match ip address prefix-list secondblock
 set community no-export
route-map routerD-out permit 20
```

Public AS

- **Router C Configuration**

```
router bgp 100
 neighbor 222.222.10.1 remote-as 130
 neighbor 222.222.10.1 default-originate
 neighbor 222.222.10.1 prefix-list Customer in
 neighbor 222.222.10.1 prefix-list default out
!
ip prefix-list Customer permit 221.10.0.0/19 le 21
ip prefix-list default permit 0.0.0.0/0
```

Loadsharing to the same ISP

Cisco.com

- **Router D Configuration**

```
router bgp 100
 neighbor 222.222.10.5 remote-as 130
 neighbor 222.222.10.5 default-originate
 neighbor 222.222.10.5 prefix-list Customer in
 neighbor 222.222.10.5 prefix-list default out
!
ip prefix-list Customer permit 221.10.0.0/19 le 21
ip prefix-list default permit 0.0.0.0/0
```

Loadsharing to the same ISP

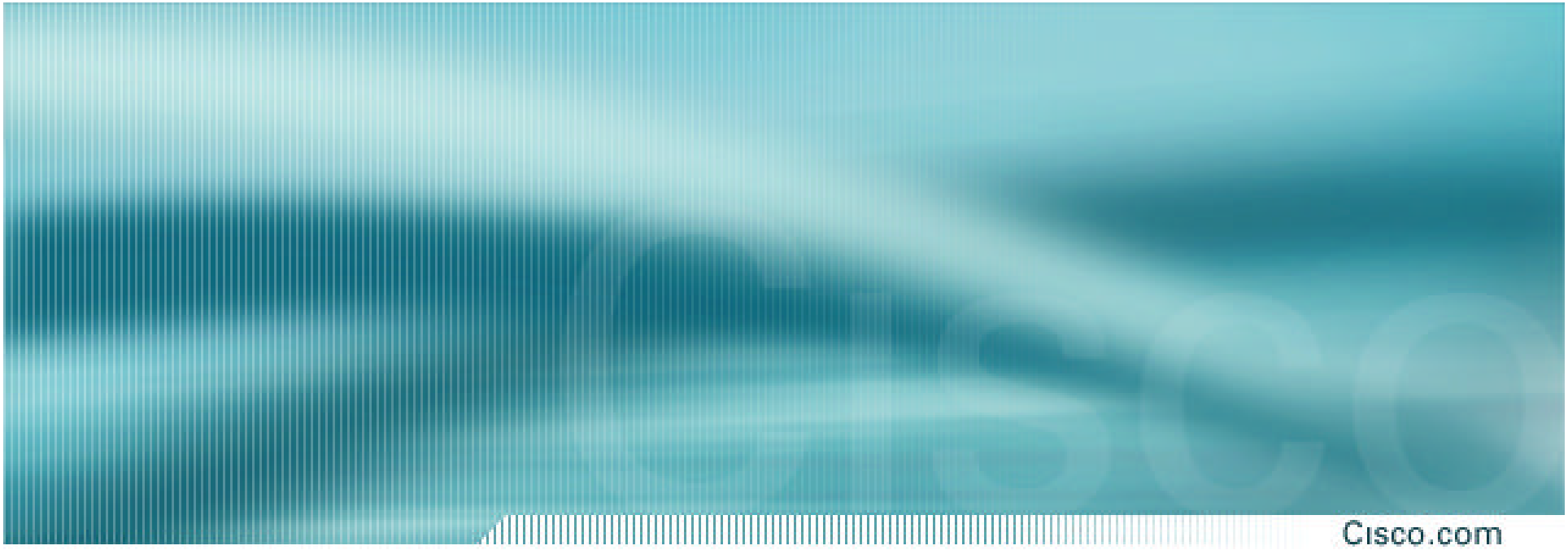
- **Router E Configuration**

```
router bgp 100
 neighbor 222.222.10.17 remote-as 110
 neighbor 222.222.10.17 filter-list 1 out
!
ip as-path access-list 1 permit ^130$
ip as-path access-list 1 permit ^$
```

- **Router E only has to announce AS130 in the same way it announces other ASes**

Public AS

- AS100 routers will not advertise prefixes marked with community no-export to other ASes
- AS100 ISP has no configuration work to do
  - AS130 ISP can control his own loadsharing
- Only a single /19 prefix is announced to the Internet - no routing table bloat! :-)



Enterprise Multihoming

Enterprise Multihoming

Cisco.com

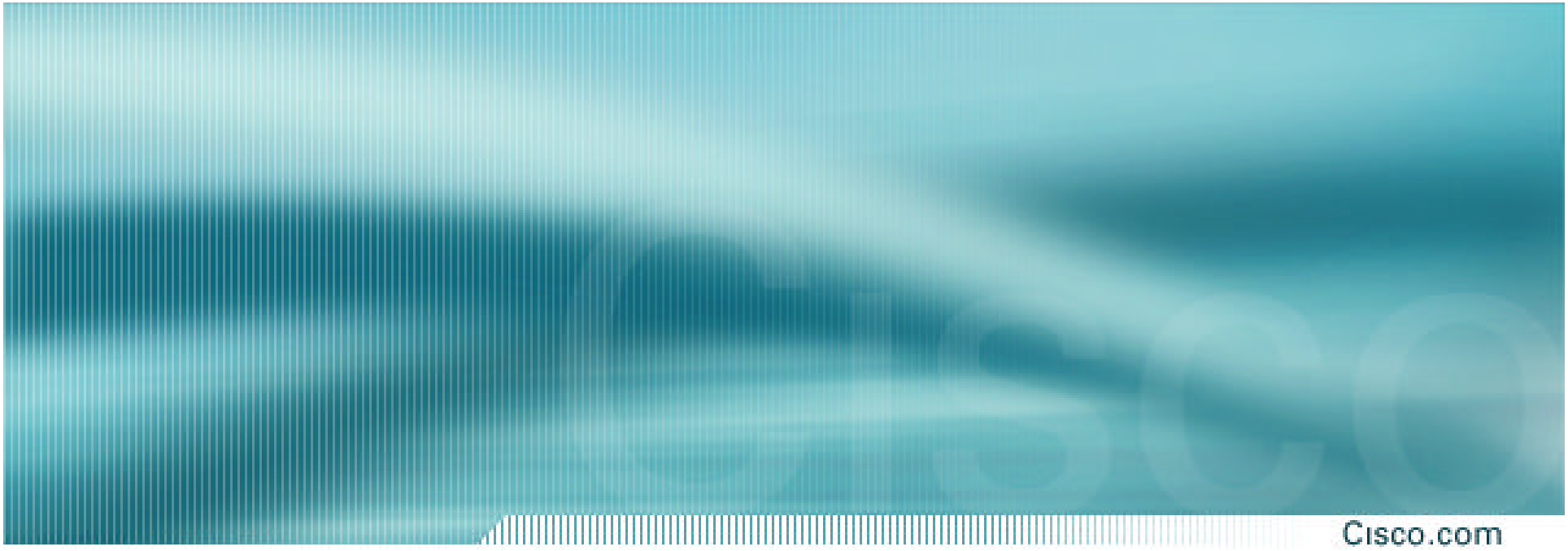
- Common scenario in Internet today
- More and more non-SPs multihoming for:
  - service provider redundancy
  - link redundancy
- Issues on Internet today:
  - Routing Table size accelerating
  - more and more /24 prefixes appearing in Internet Routing Table
  - ASN consumption accelerating

Enterprise Multihoming

- **The following examples**
  - **apply to smaller ISPs who don't yet have their own address block**
  - **require BGP but a private AS (ASN >64511) can and should be used**
  - **are good for the health of the Internet**

Medium/Large ISP Multihoming

- **ISPs should obtain their own address block and ASN**
  - **Get it from RIR**
  - **Makes multihoming easier**
  - **Makes changing upstreams easier**
  - **Does not cause so much fragmentation in Internet Routing Table**



Enterprise Multihoming

Example One

Provider Redundancy

Enterprise Multihoming

- **Common situation is enterprise multihoming**
  - **address space used by enterprise comes from both upstream ISPs**
  - **multihoming and loadsharing more difficult**
  - **want to avoid leaking subprefixes of upstream provider address space when possible**
  - **require provider redundancy (not just link redundancy)**

Enterprise Multihoming

- **Address space from upstream should match link bandwidth to upstream, e.g.**

  ISP1 ® Enterprise = 256kbps ® /22

  ISP2 ® Enterprise = 128kbps ® /23

  assumes address space is uniformly distributed across network

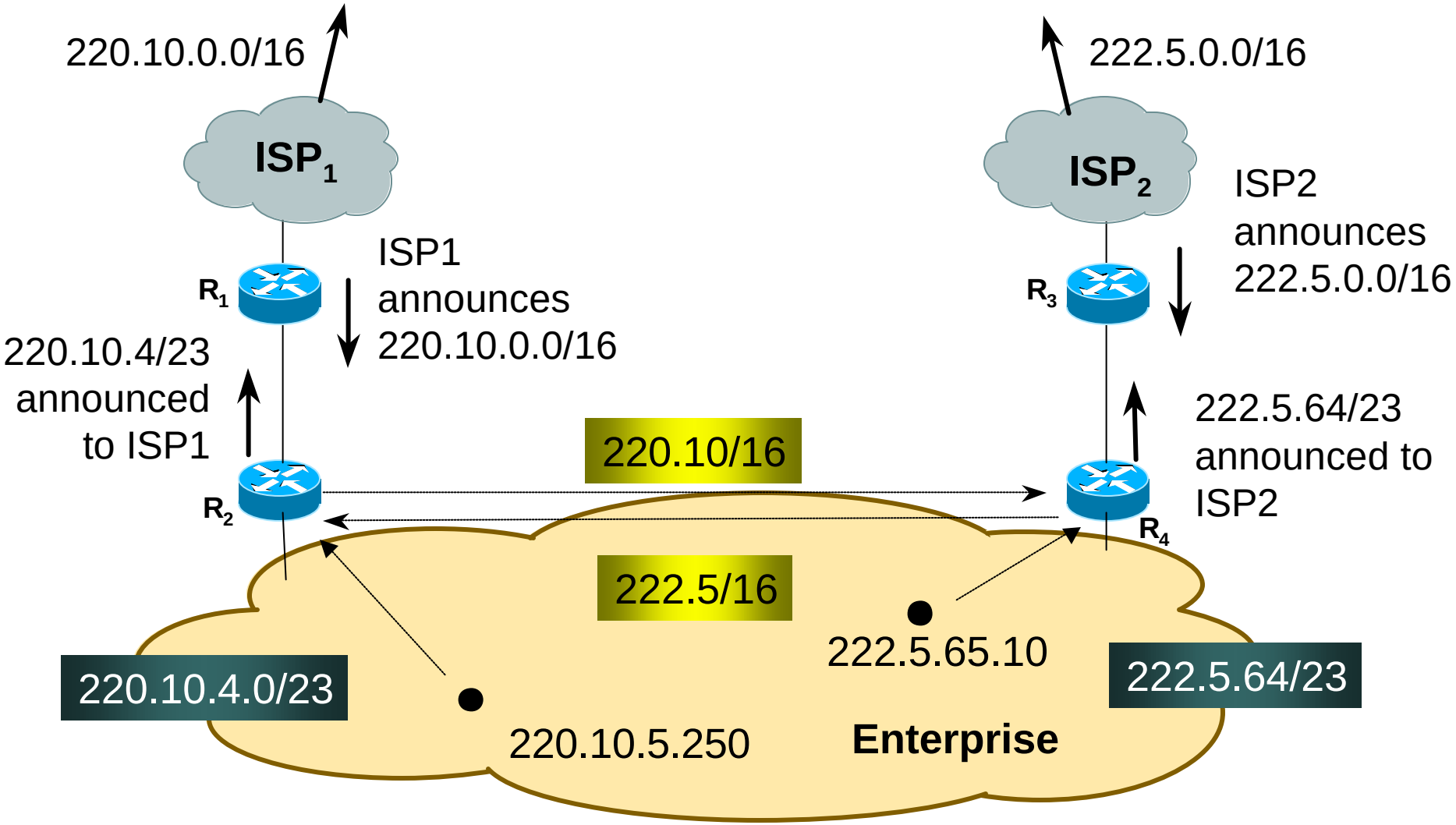
  assumes that there is a requirement for 3x /23 in the Enterprise backbone

- **Next example assumes equal bandwidth links from Enterprise to ISP1 and ISP2**

Enterprise Multihoming Conditional Advertisement

- **Conditional advertisement feature in BGP**
  - **loadsharing under normal conditions**
  - **subprefixes only announced in failure scenarios**
  - **requires upstreams to announce only one prefix to enterprise border network**

Steady State



Steady State

- ISP1 has 220.10.0.0/16 address block
- ISP2 has 222.5.0.0/16 address block
- Enterprise customer multihomes
  - upstreams don't announce subprefixes
  - can use private AS (ASN>64511)
  - R2 and R4 originate default in their IGP
    - outbound traffic uses nearest exit (IGP metrics)

Steady State

- **Router2 configuration:**

```
router bgp 65534
 network 220.10.4.0 mask 255.255.254.0
 network 222.5.64.0 mask 255.255.254.0
 neighbor <R1> remote-as 150
 neighbor <R1> prefix-list isp1-in in
 neighbor <R1> prefix-list isp1-out out
 neighbor <R1> advertise-map isp2-sb non-exist-map isp2-bb
 neighbor <R4> remote-as 65534
 neighbor <R4> update-source loopback 0
!
ip route 220.10.4.0 255.255.254.0 null0 250
..next slide
```

Steady State


```
ip route 222.5.64.0 255.255.254.0 null0 250
!
ip prefix-list isp1-out permit 220.10.4.0/23
ip prefix-list isp2-out permit 222.5.64.0/23
!
ip prefix-list isp1-in permit 220.10.0.0/16
ip prefix-list isp2-in permit 222.5.0.0/16
!
route-map isp2-sb permit 10
 match ip address prefix-list isp2-out
!
route-map isp2-bb permit 10
 match ip address prefix-list isp2-in
!
```

Steady State

- **Router2 peers iBGP with Router4**

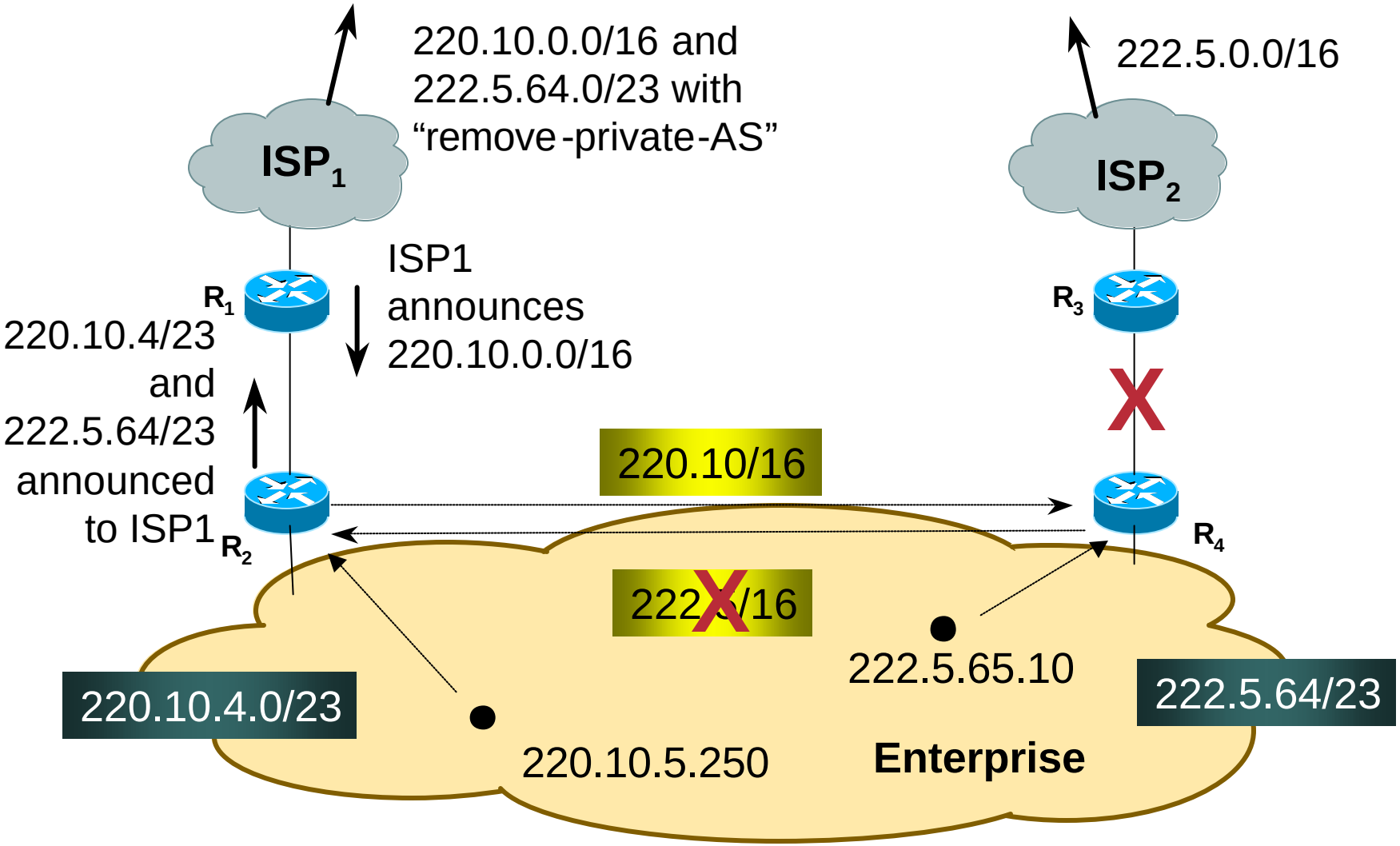
  hears ISP2's /16 prefix

- **Router2 peers eBGP with Router1**

  hears ISP1's /16 prefix only

  announces 220.10.4.0/23 only

Link Failure



Link Failure

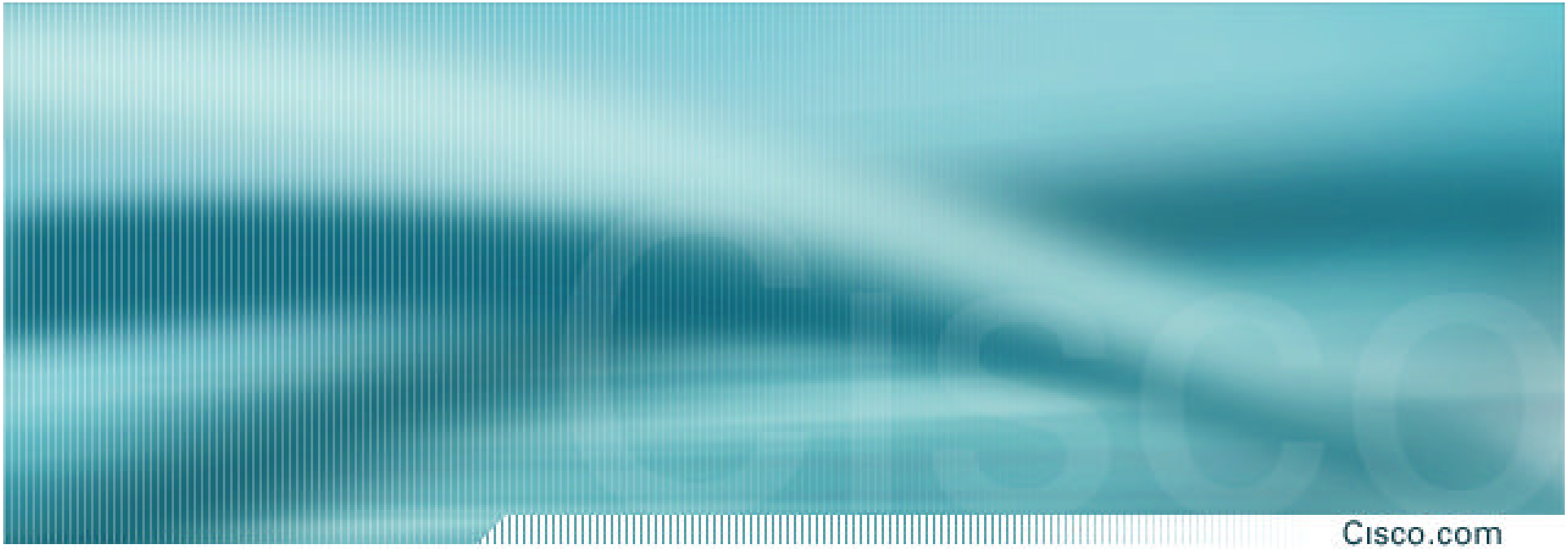
- **Peering between Router 4 and Router3 (ISP2) goes down**
  - **222.5.0.0/16 prefix withdrawn**
- **Conditional advertisement process activated**
  - **Router2 starts to announce 222.5.64.0/23 to Router1**
- **Connectivity for Enterprise maintained**

Enterprise Multihoming

- **Conditional advertisement useful when address space comes from both upstreams**
  - **no subprefixes leaked to Internet unless in failure situation**
- **Alternative backup mechanism would be to leak /23 prefixes with longer AS path**
  - **routing table bloat, reachability issues**

What goes in the Internet Routing Registry?

- ISP1 and ISP2 obviously put their own address blocks as route objects in the IRR
- ISP1 will put the ISP1 subprefix which Enterprise will announce into the IRR with origin-as of ISP2
- ISP2 will put the ISP2 subprefix which Enterprise will announce into the IRR with origin-as of ISP1
- No inconsistent origin AS, no “problem”



Enterprise Multihoming

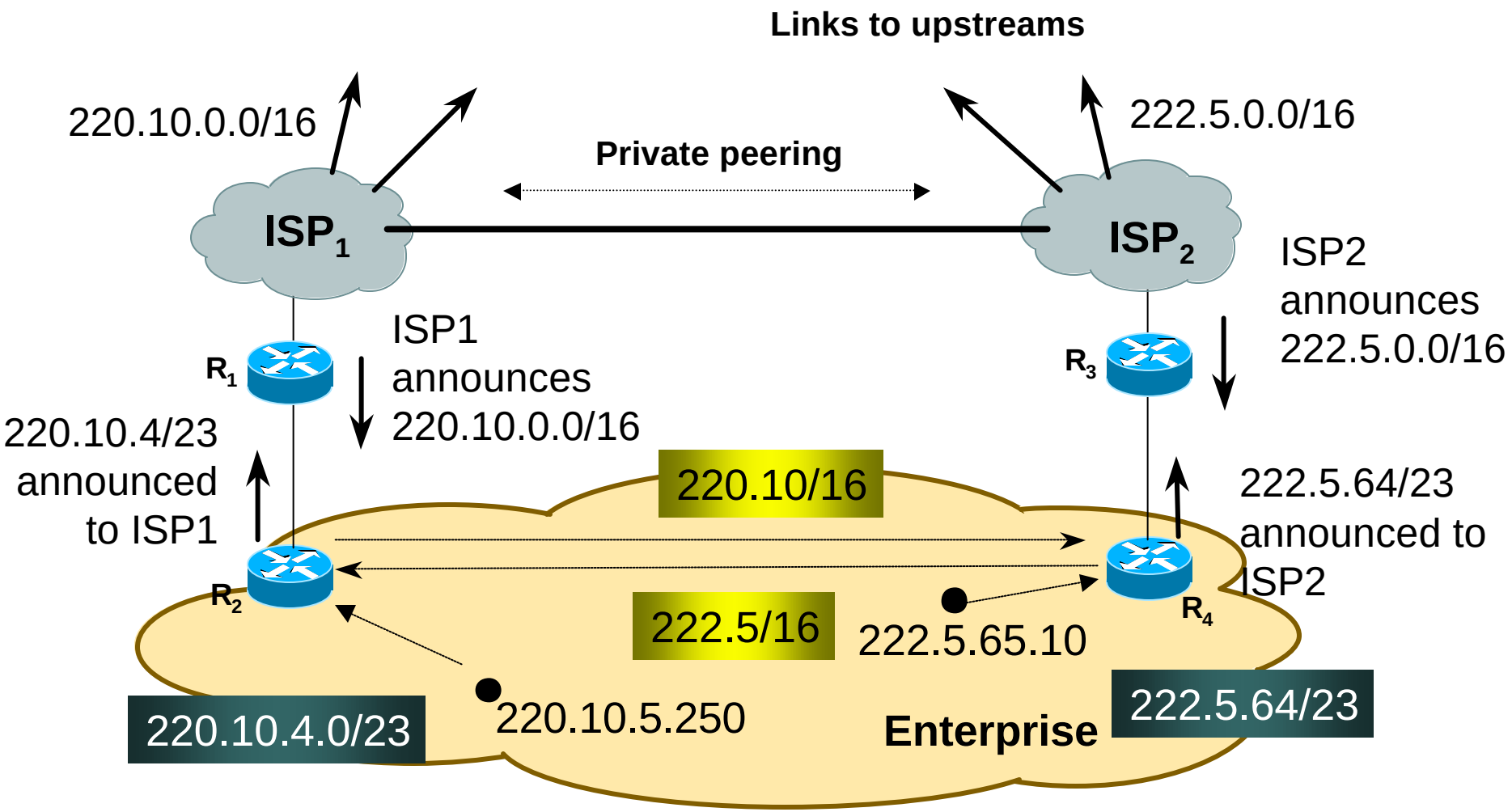
Example Two

Link Redundancy

Enterprise Multihoming

- **Situation similar to previous example**
  - **address space used by enterprise comes from both upstream ISPs**
  - **use conditional advertisement**
  - **want to avoid leaking subprefixes of upstream provider address space into the Internet**

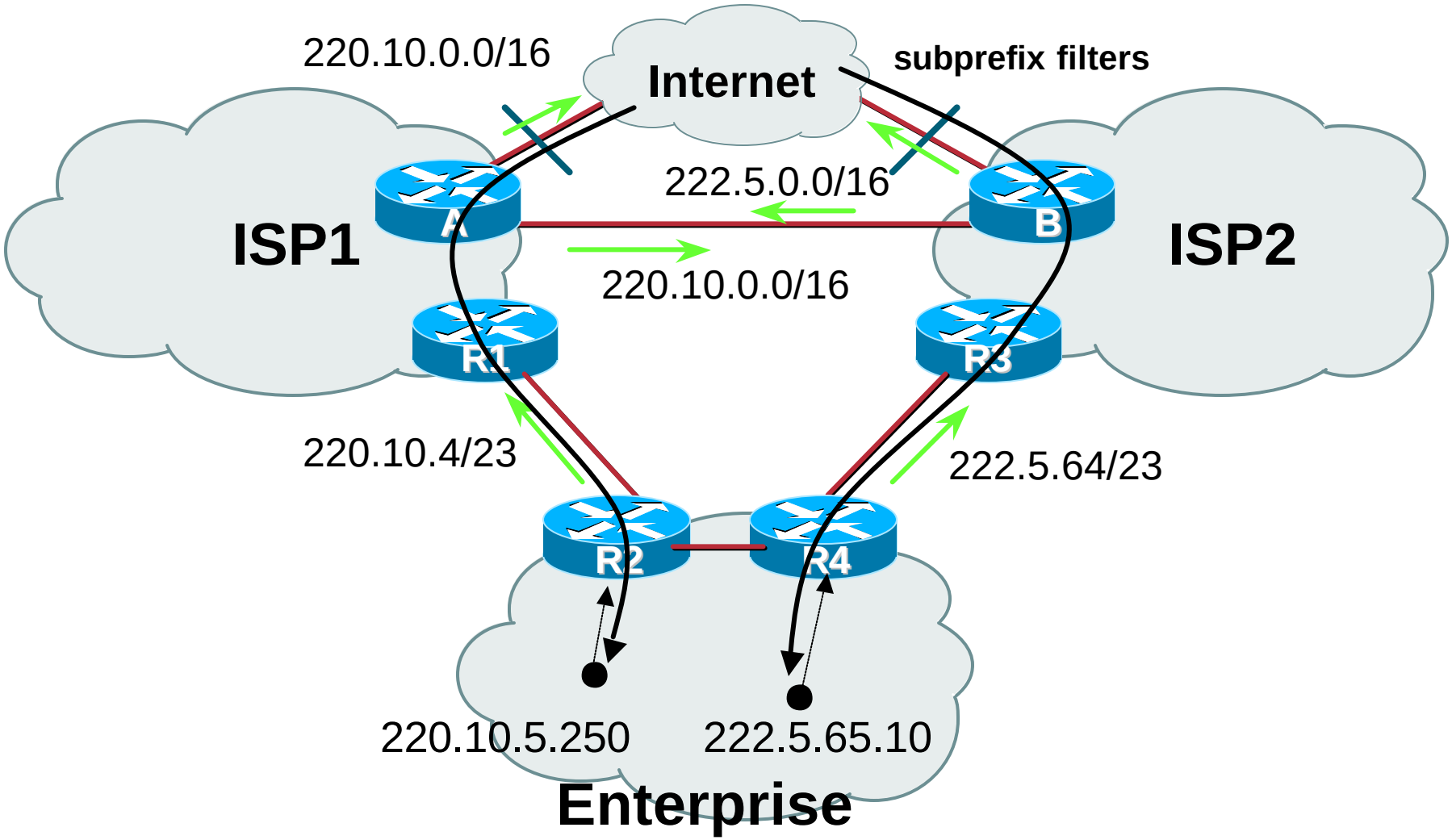
Steady State



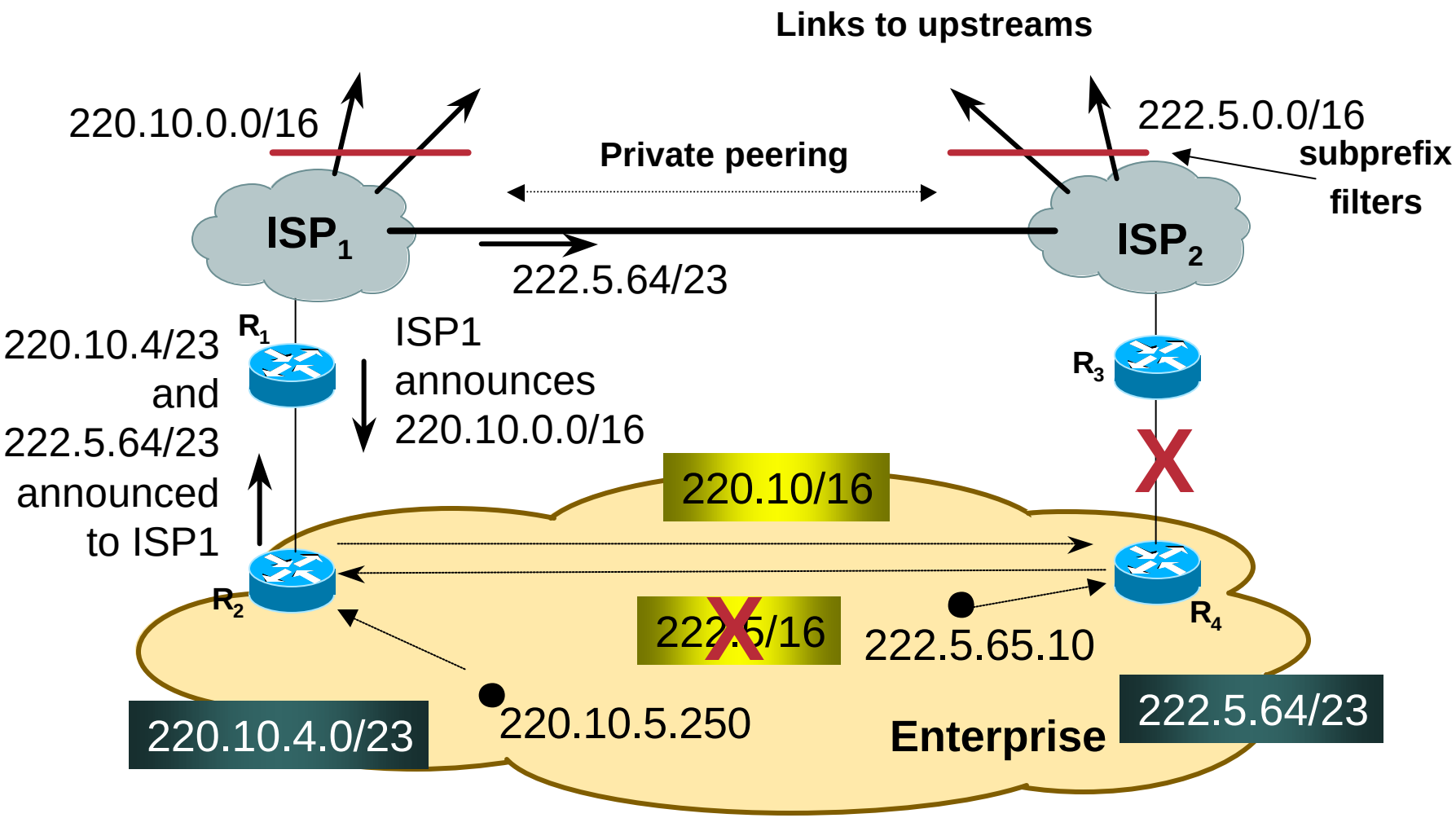
Steady State

- ISP1 and ISP2 have private peering
  - exchange each other's prefixes
  - enterprise customer is looking for link redundancy only
  - no subprefixes leaked to Internet
- Configuration of R2 as in previous example

Traffic Flow
Steady State



Link Failure



Link Failure

- **R3 ® R4 link goes down**

  conditional advertisement effective

  222.5.64/23 announced by R2 to R1

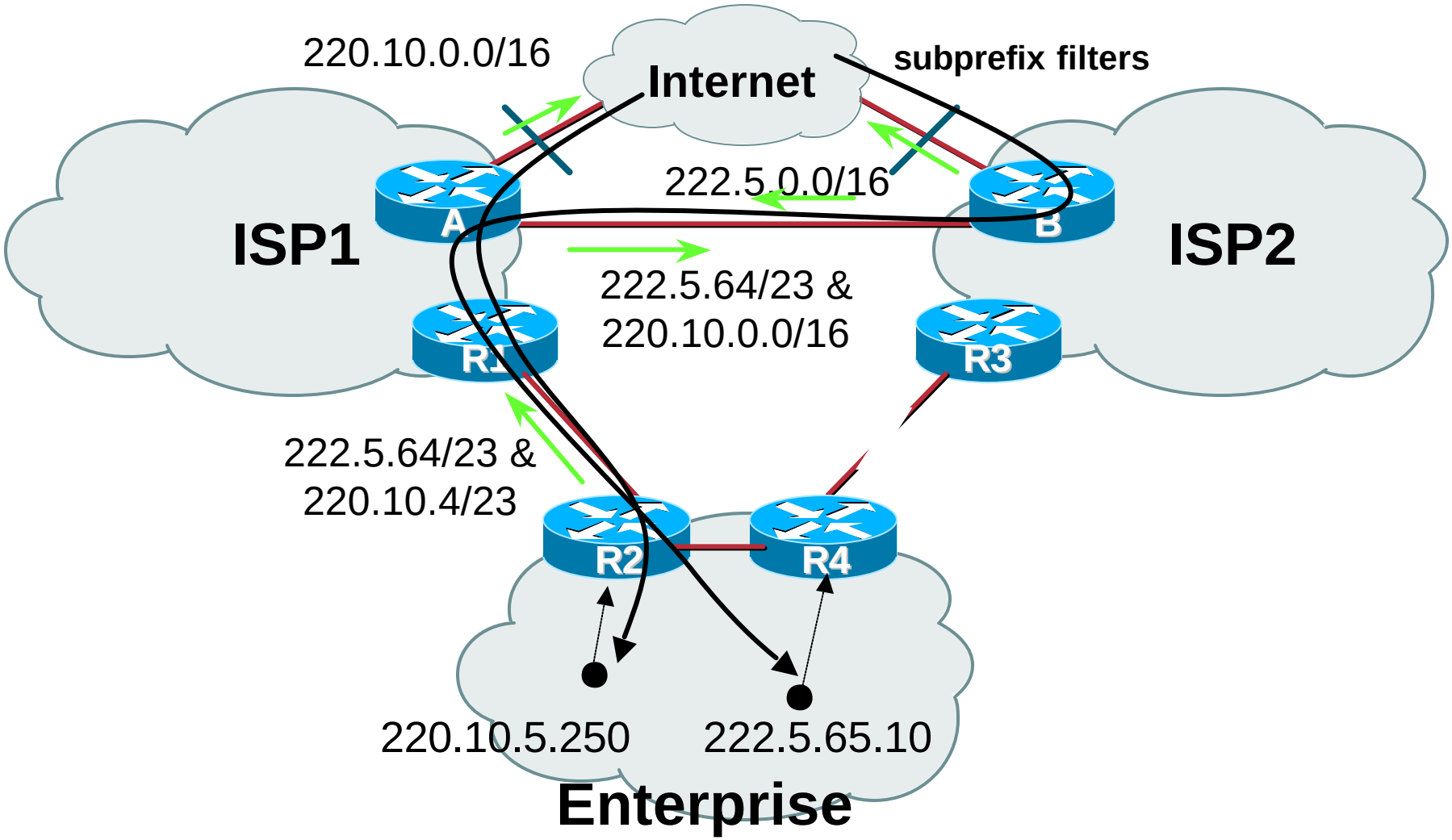
  222.5.64/23 announced by ISP1 to ISP2

- **Filters!**

  ISP1 and ISP2 filter subprefixes from their blocks outbound to Internet

  backup yet no subprefixes leaked to Internet

Link Failure



Configuration

- **RouterA ISP1 border router configuration:**

```
router bgp 150
 network 220.10.0.0 mask 255.255.0.0
 neighbor <routerB> remote-as 140
 neighbor <routerB> prefix-list isp2-in in
 neighbor <routerB> prefix-list isp2-out out
 neighbor <upstream> remote-as 110
 neighbor <upstream> prefix-list rfc1918-sua in
 neighbor <upstream> prefix-list myblock out
!
ip route 220.10.0.0 255.255.0.0 null0
..next slide
```

Configuration

Cisco.com

```
ip prefix-list isp2-out permit 220.10.0.0/16
ip prefix-list isp2-out permit 222.5.64.0/23
!
ip prefix-list isp2-in permit 222.5.0.0/16
ip prefix-list isp2-in permit 220.10.4.0/23
!
ip prefix-list myblock permit 220.10.0.0/16
!
```

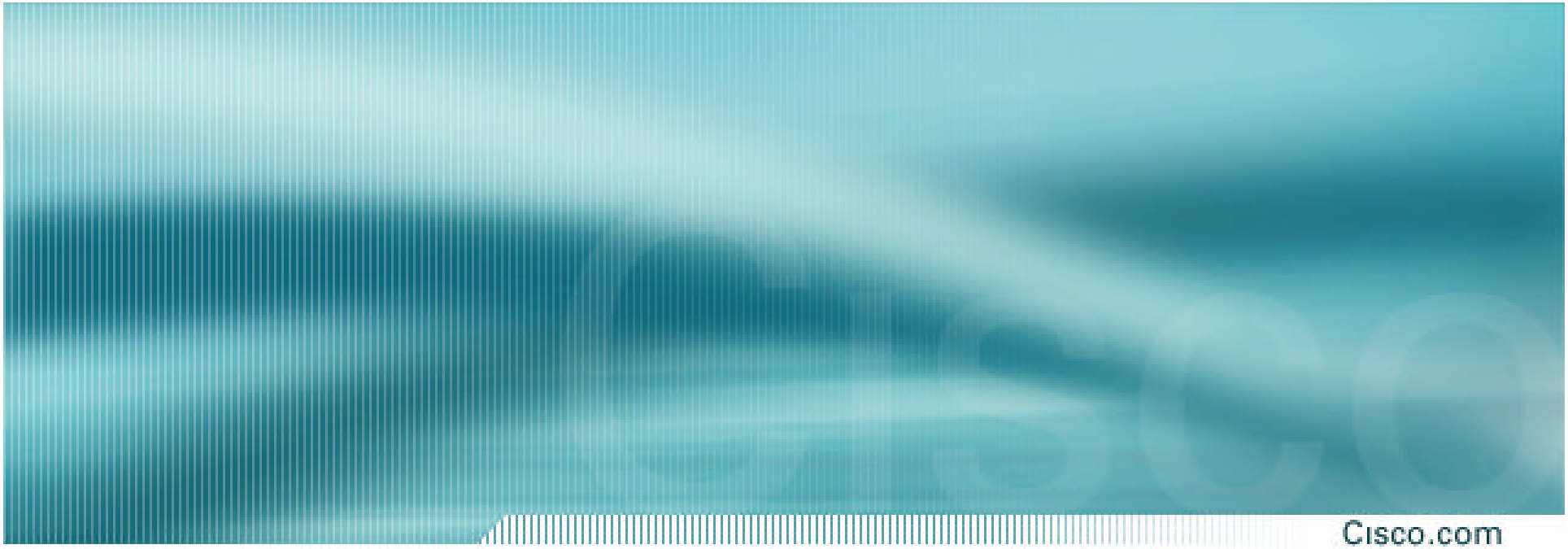
- **The “myblock” prefix list ensures that no subprefixes are leaked to the Internet routing table**

Recommendations

- **Address space for Enterprise network should be obtained from both upstreams**
  - **according to link bandwidths**
- **Address space should be distributed according to utilisation**
  - **loadsharing is about address assignment policies, monitoring bandwidth utilisation, as well as BGP attribute manipulation**
- **Use a private AS – no need for a public AS**
  - **needs agreement between two upstreams**

What goes in the Internet Routing Registry?

- ISP1 and ISP2 obviously put their own address blocks as route objects in the IRR
- No need for any other entries as no subprefixes appear in the global internet routing table
- No inconsistent origin AS, no “problem”



BGP and the Internet

Multihoming