Route Servers for !Dummies

or: Scaling is Hard; Let's Go Shopping!

inex

internet neutral exchange

Nick Hilliard

Head of Operations

nick@inex.ie

Some Blurb on INEX

- Currently only member-owner IXP in Ireland
- 59 members, 46 full members, 13 associate
- Estimate about 90% eyeballs in Ireland (South)
- Traffic levels: daytime peaks of 6G
- Provide usual services - 10M to 10G ethernet
- Two separate L2 infrastructures
- Three PoPs: Telecity Dublin, DEG, Interxion DUB1
- Mixture of Brocade (FES-X624, TI24X) and Cisco 6500
- Fibre lit with Transmode DWDM kit - N x 10G
- Highly active community interest

Some Blurb on INEX

- Currently only member-owner IXP in Irela
- 59 members, 46 full members, 1
- Estimate about 90% eye
- Traffic levels: d
- Provid G ethernet
- ures
- ty Dublin, DEG, Interxion DUB1
- Brocade (FES-X624, TI24X) and Cisco 6500
- bre lit with Transmode DWDM kit - N x 10G
- Highly active community interest

Free Beer at Meetings !!!11!!

Some Blurb on INEX

- Currently only member-owner IXP in Ireland
- 59 members, 46 full members, 13 associate
- Estimate about 90% eyeballs in Ireland (South)
- Traffic levels: daytime peaks of 6G
- Provide usual services - 10M to 10G ethernet
- Two separate L2 infrastructures
- Three PoPs: Telecity Dublin, DEG, Interxion DUB1
- Mixture of Brocade (FES-X624, TI24X) and Cisco 6500
- Fibre lit with Transmode DWDM kit - N x 10G
- Highly active community interest
- Oh yeah, we have some route-servers too

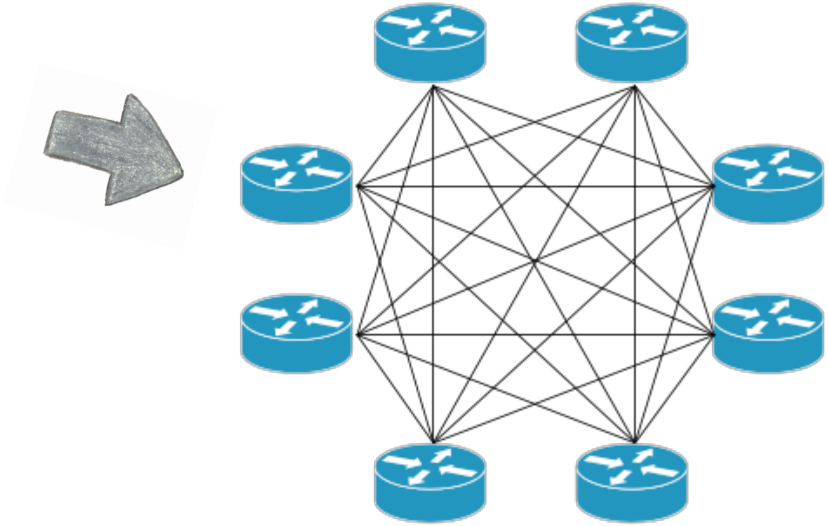
Route Servers for Dummies

- Platform for multi lateral peering agreements (MPLA)
- Similar to a route reflector, except uses eBGP
- Very fashionable at IXPs right now
  - Reduce administrative load of peering
  - Simple interconnection to lots of other partners
  - Instant RoI (ISP management likes this)
  - Outsourcing RIB calculations to fast machines(!)
  - “Safe” if IXP has implemented prefix filtering
- Considered ghetto routing by larger providers
  - There are good reasons for this opinion
  - INEX recommends peering with route servers unless you know why you shouldn’t
  - Because route servers are not for everyone
- Route prefix filtering considered indispensable by IXP participants

Route Servers for Dummies

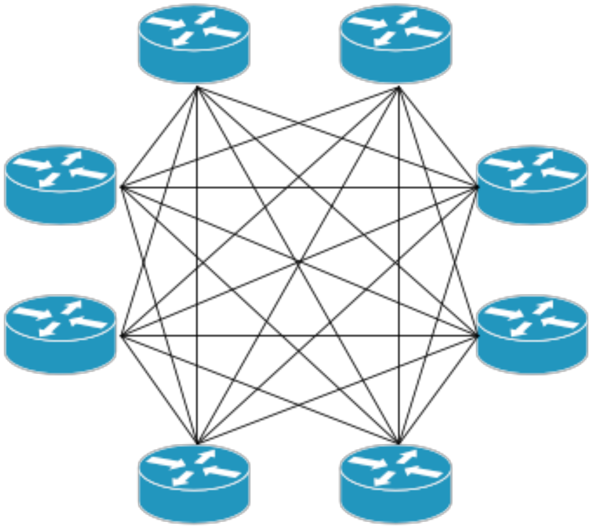
Route Servers for Dummies

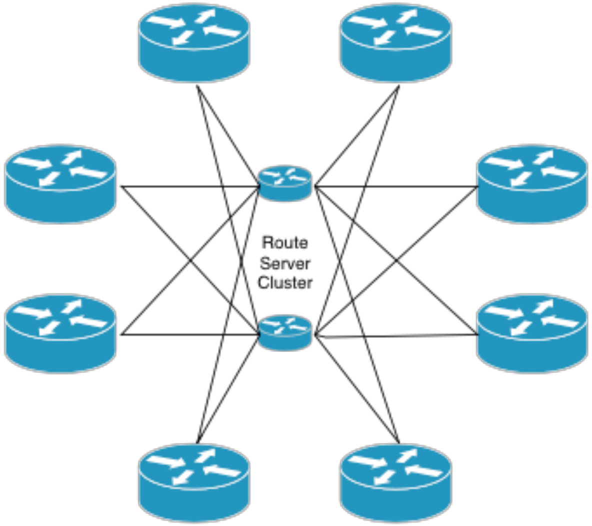
Peering on IXP without Route Servers



Route Servers for Dummies

Peering on IXP without Route Servers

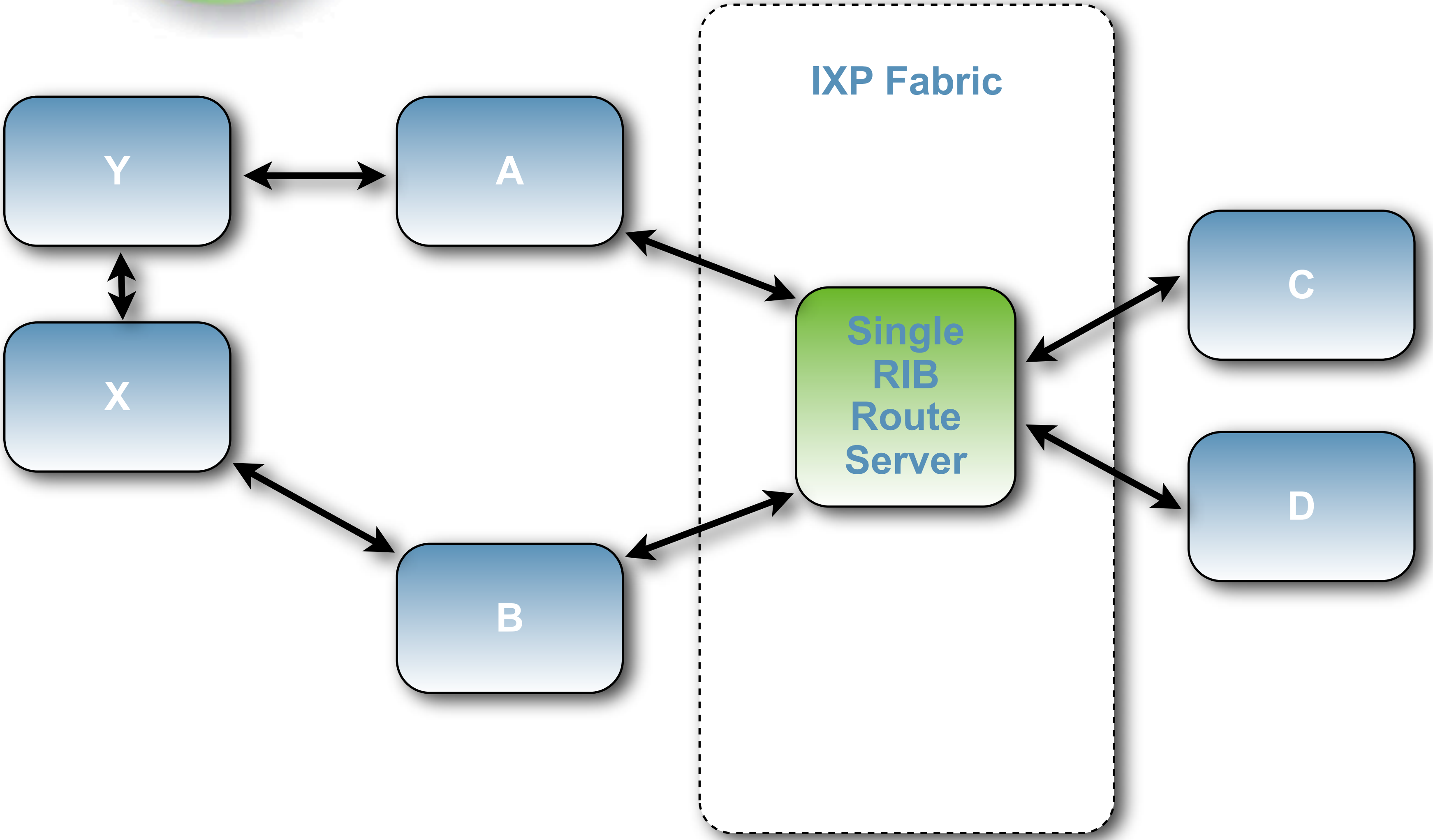




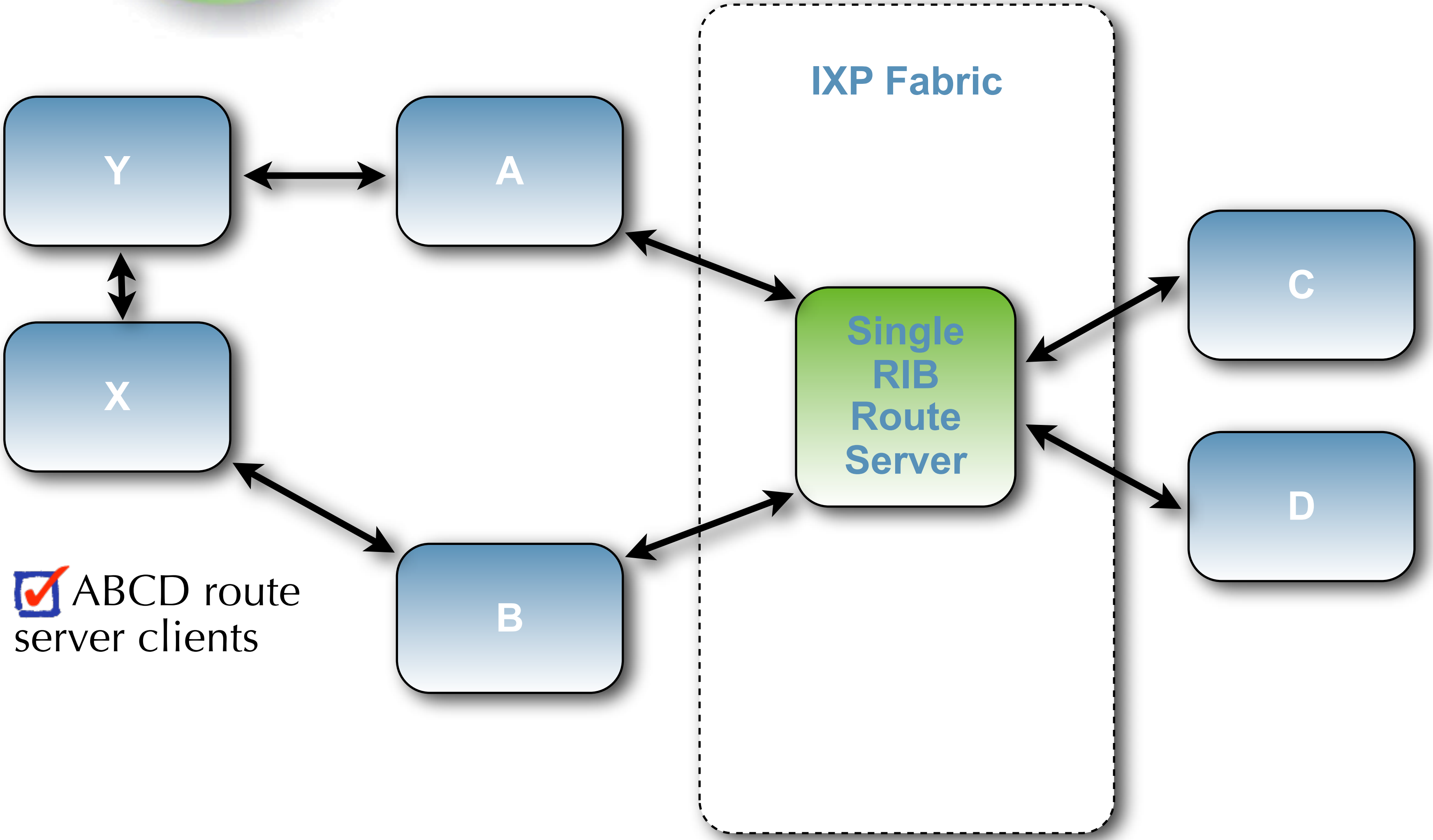
Peering on IXP with Route Servers

inex
internet neutral exchange

Single-RIB BGP policy problem

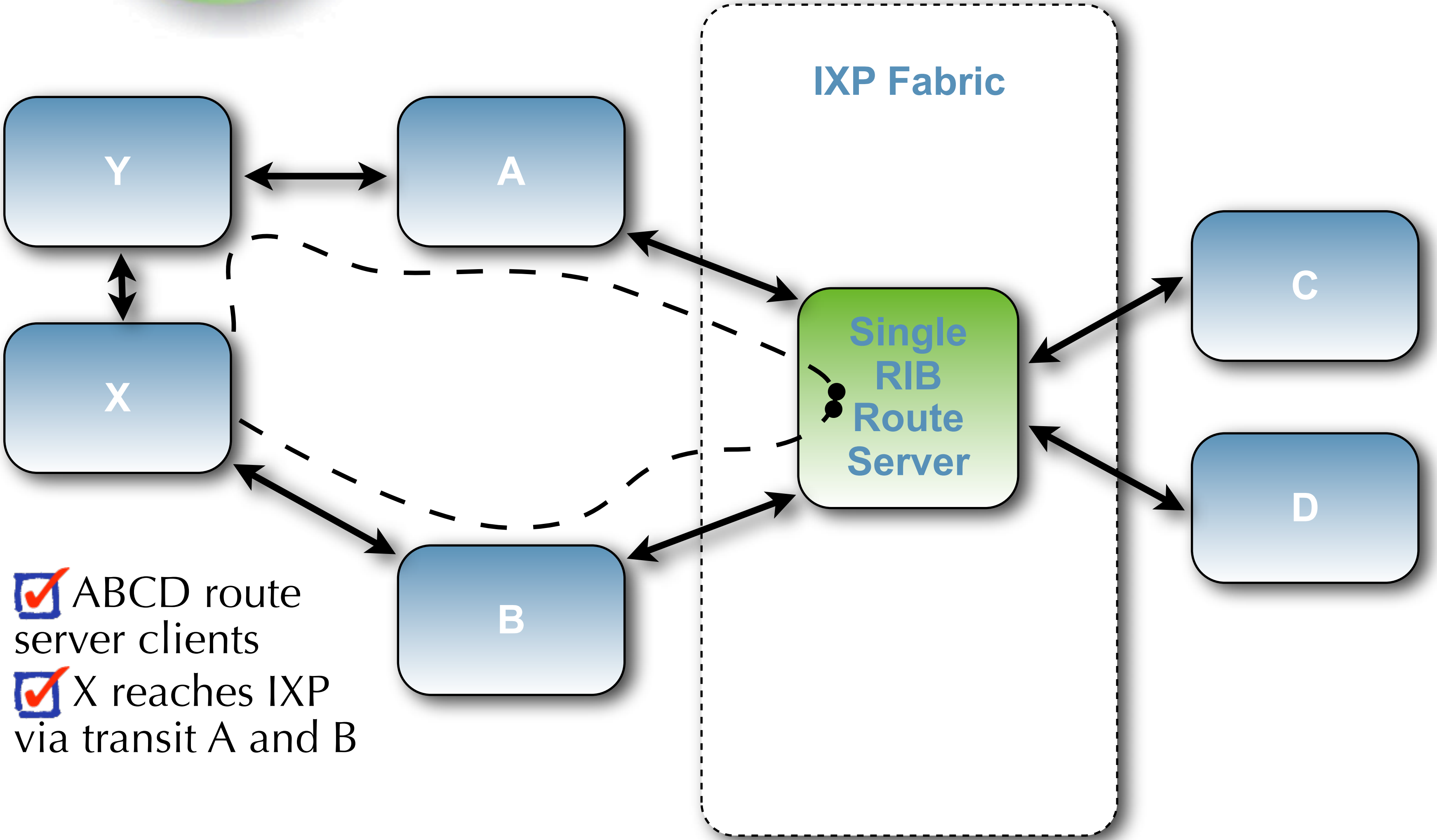


Single-RIB BGP policy problem



☑ ABCD route server clients

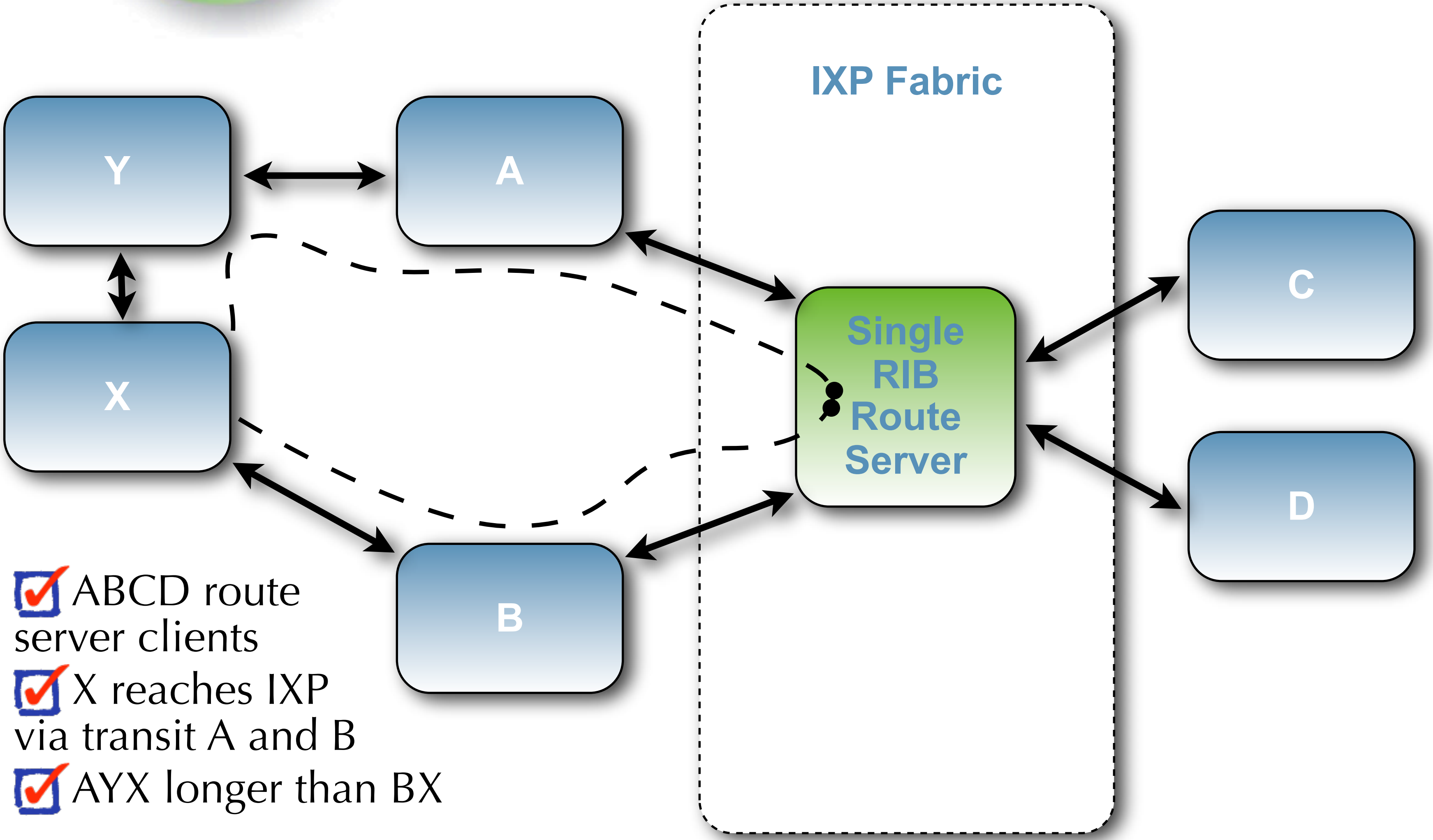
Single-RIB BGP policy problem



☑ ABCD route server clients

☑ X reaches IXP via transit A and B

Single-RIB BGP policy problem

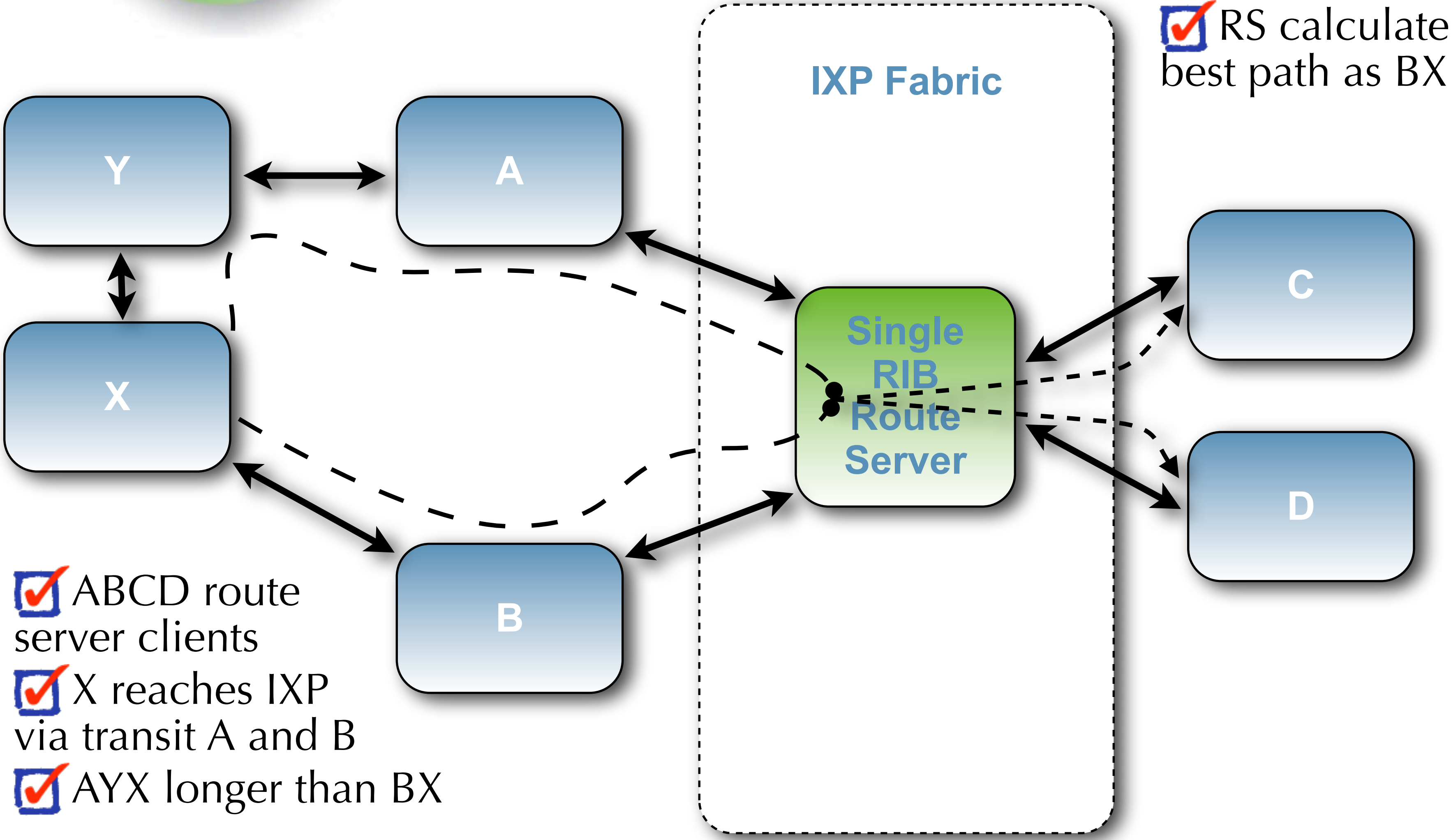


☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

Single-RIB BGP policy problem



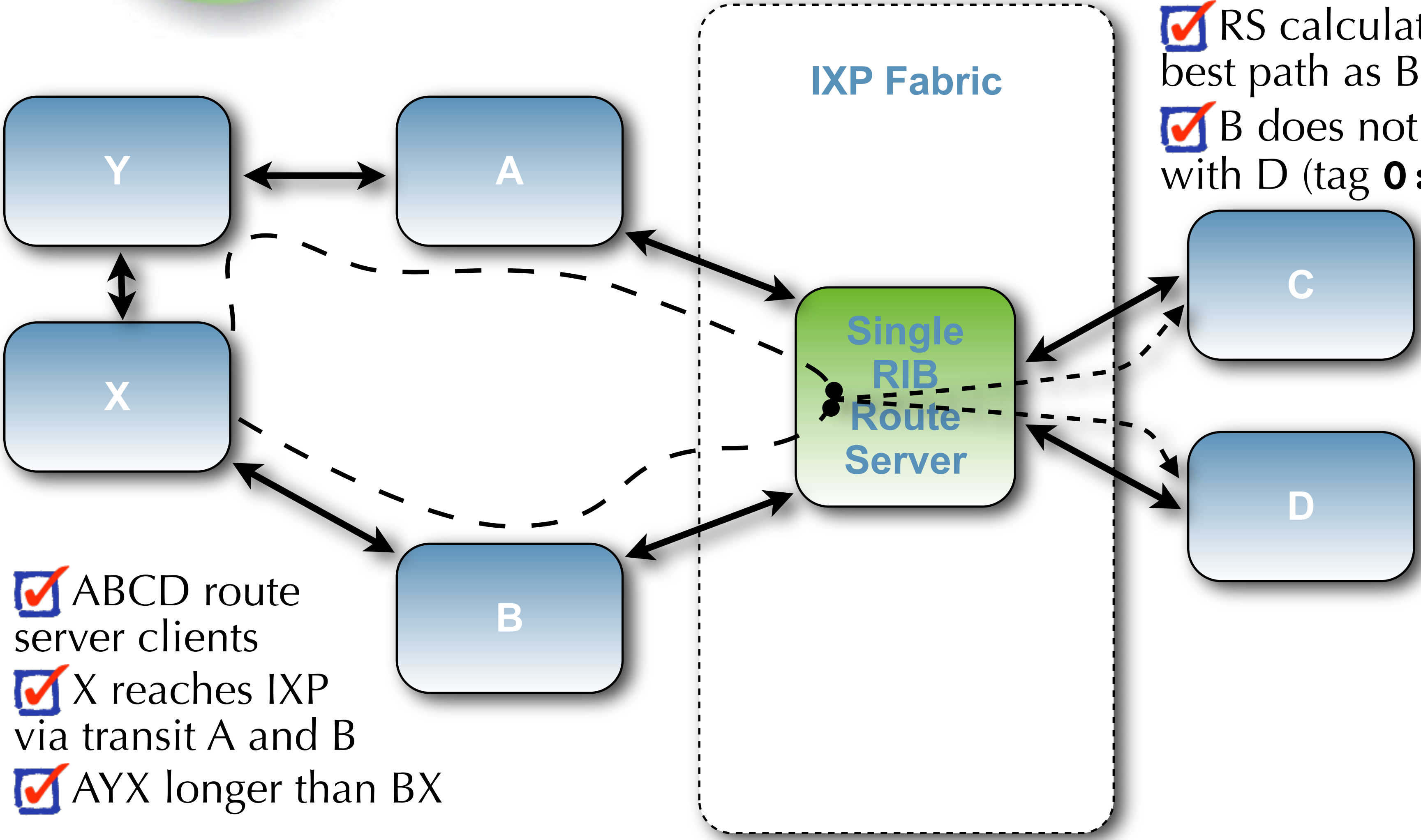
☑ RS calculates best path as BX

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

Single-RIB BGP policy problem



☑ RS calculates best path as BX

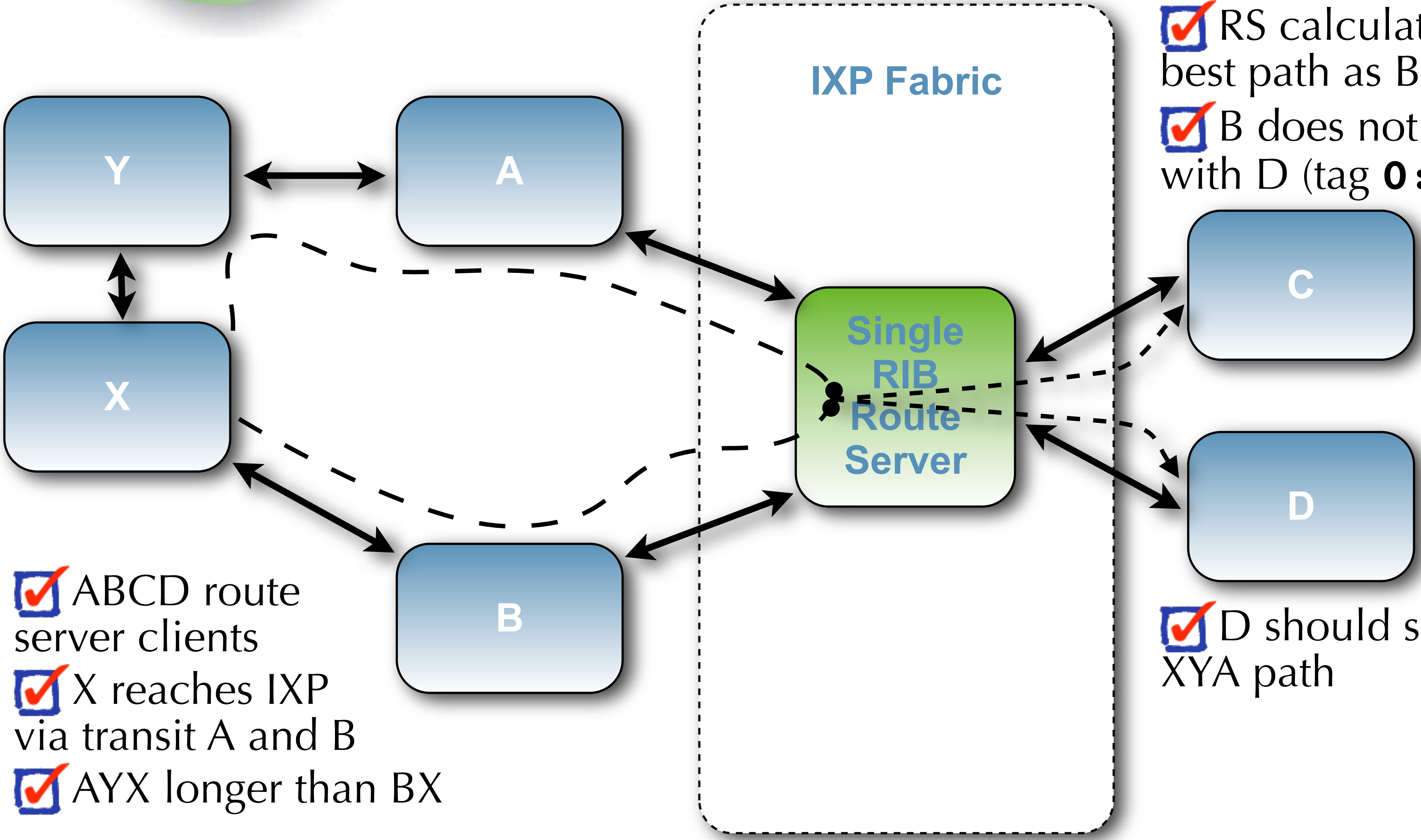
☑ B does not peer with D (tag **0:D**)

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

Single-RIB BGP policy problem



☑ RS calculates best path as BX

☑ B does not peer with D (tag **0:D**)

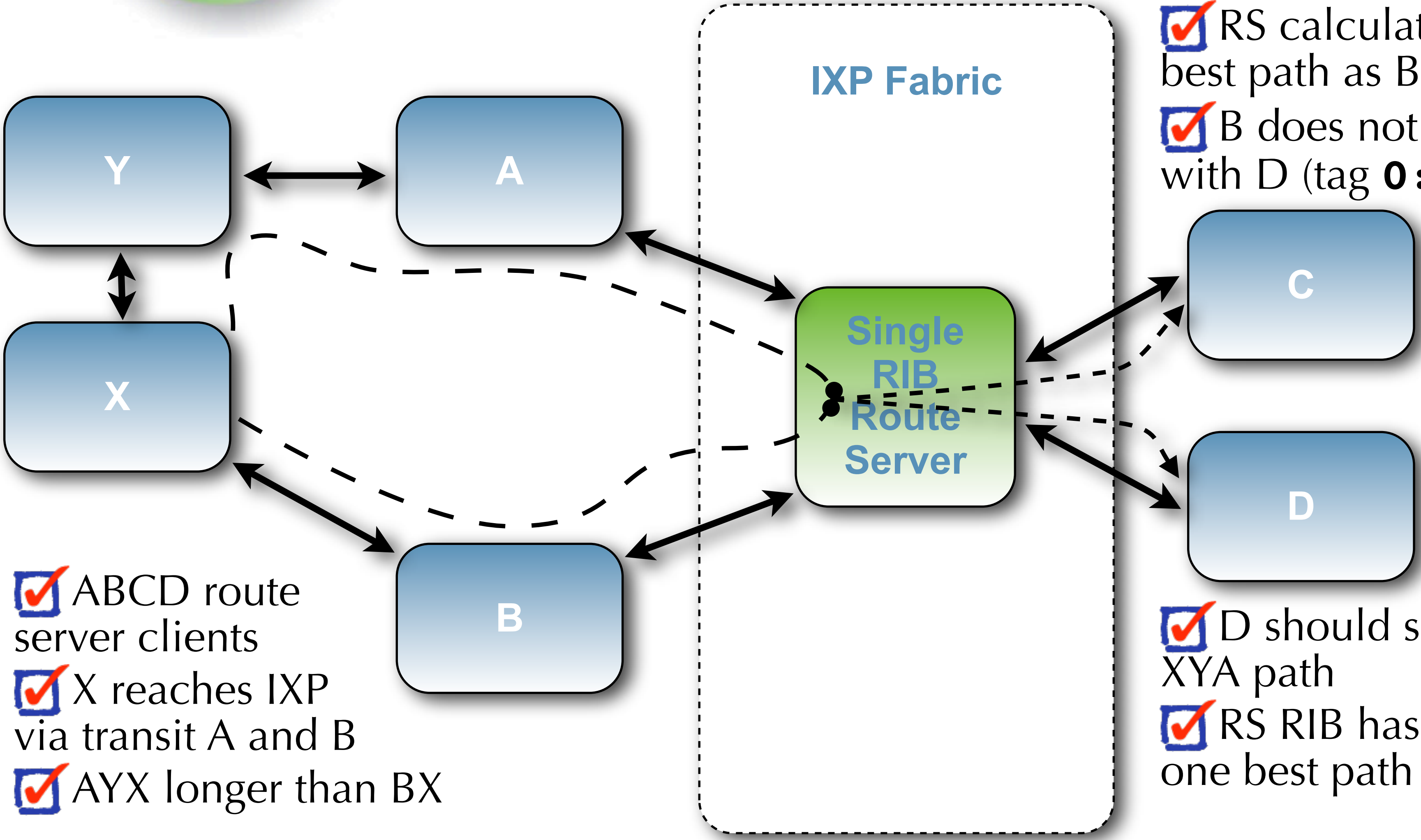
☑ D should see XYA path

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

Single-RIB BGP policy problem



☑ RS calculates best path as BX

☑ B does not peer with D (tag **0:D**)

☑ ABCD route server clients

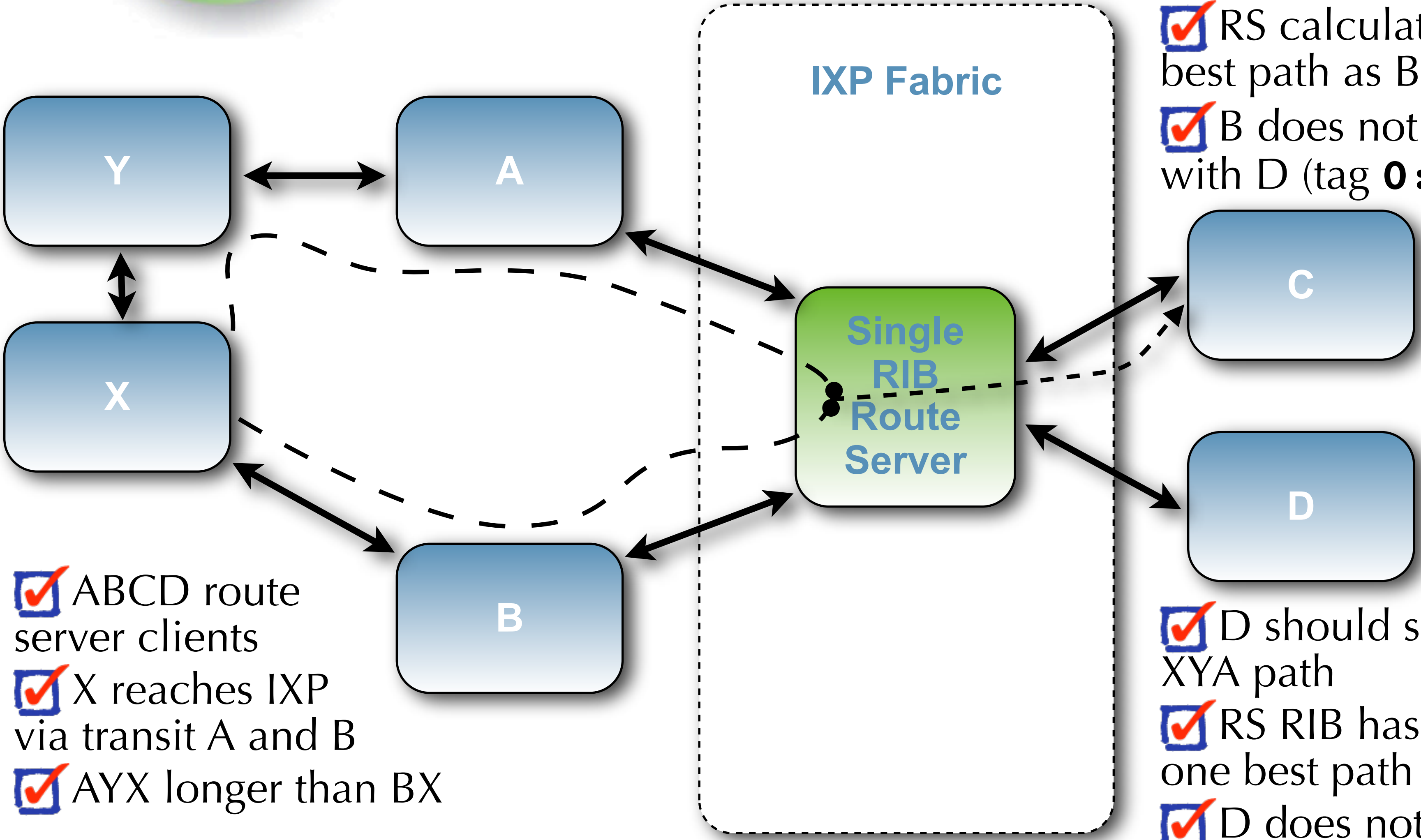
☑ X reaches IXP via transit A and B

☑ AYX longer than BX

☑ D should see XYA path

☑ RS RIB has only one best path - BX

Single-RIB BGP policy problem



☑ RS calculates best path as BX

☑ B does not peer with D (tag **0:D**)

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

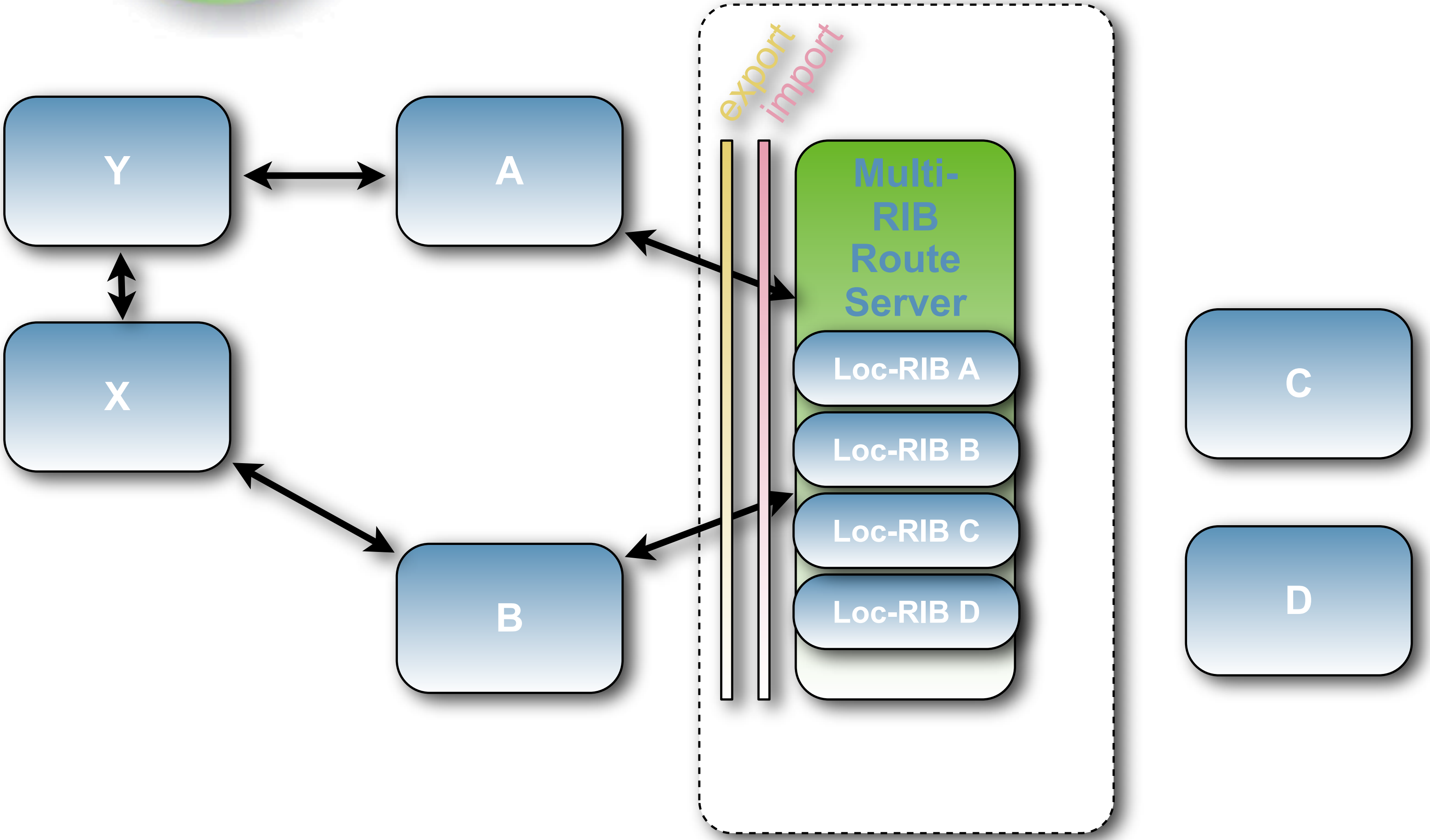
☑ AYX longer than BX

☑ D should see XYA path

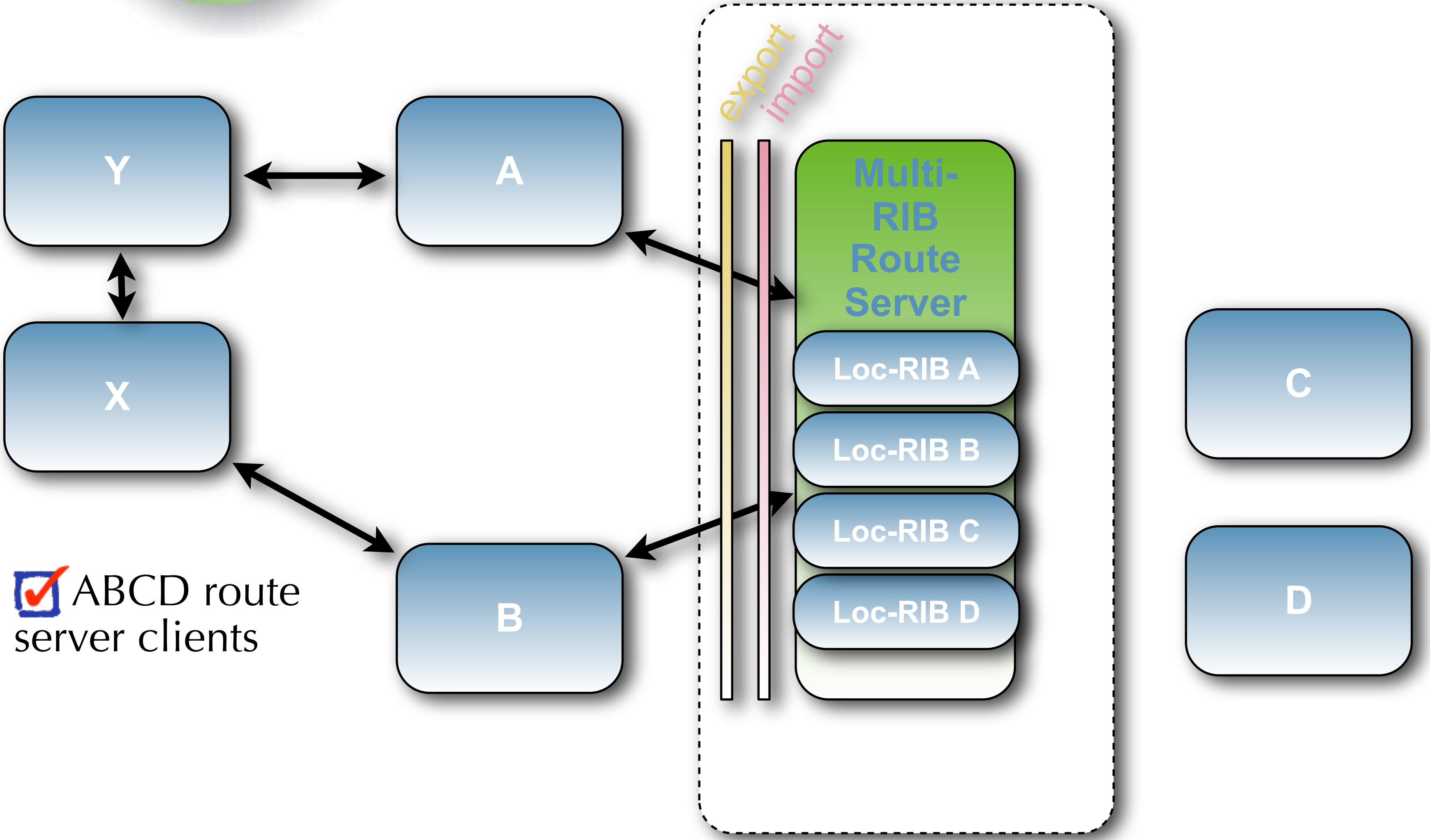
☑ RS RIB has only one best path - BX

☑ D does not see X

How Per-Client Loc-RIBs Work

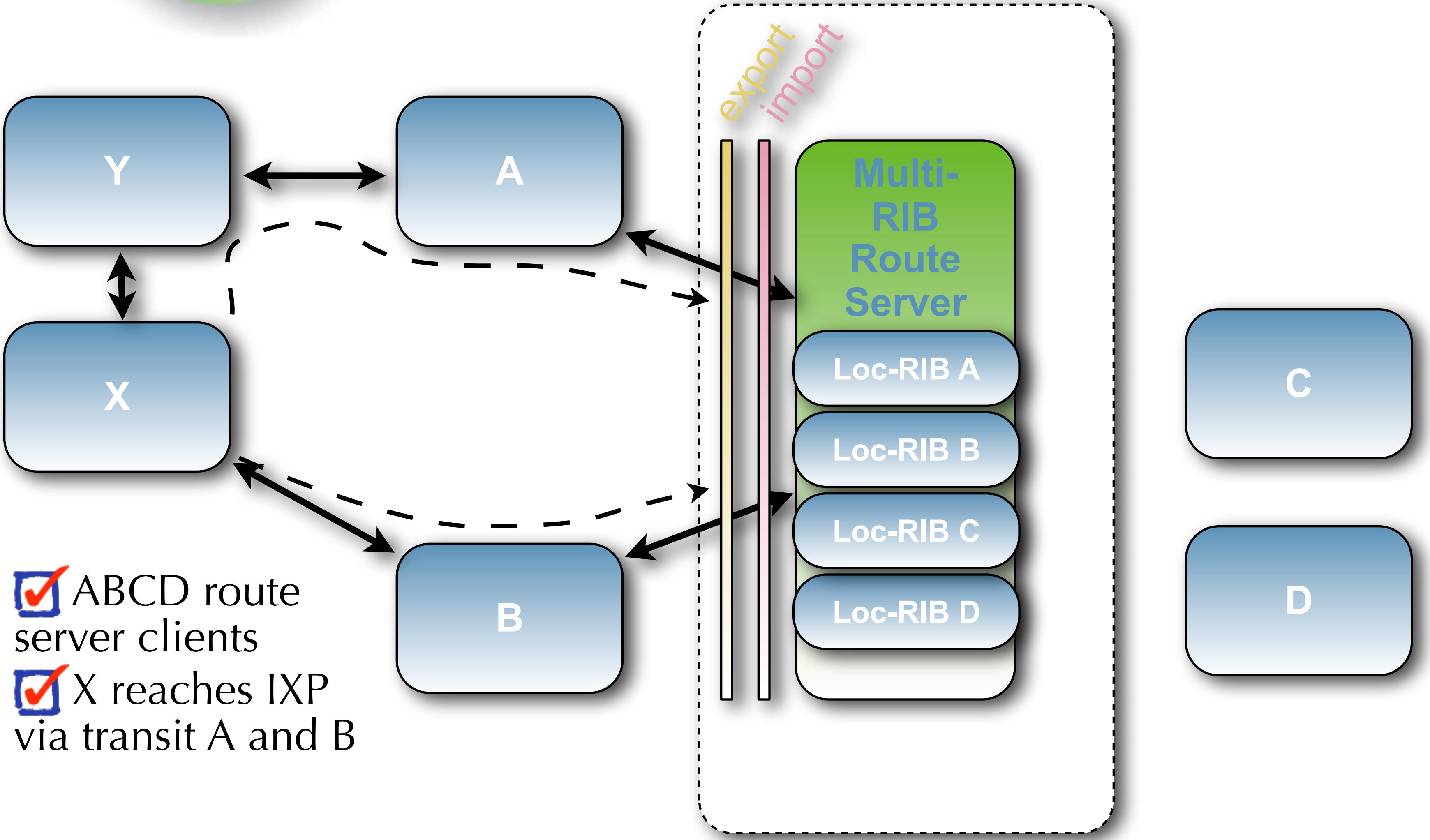


How Per-Client Loc-RIBs Work



☑ ABCD route server clients

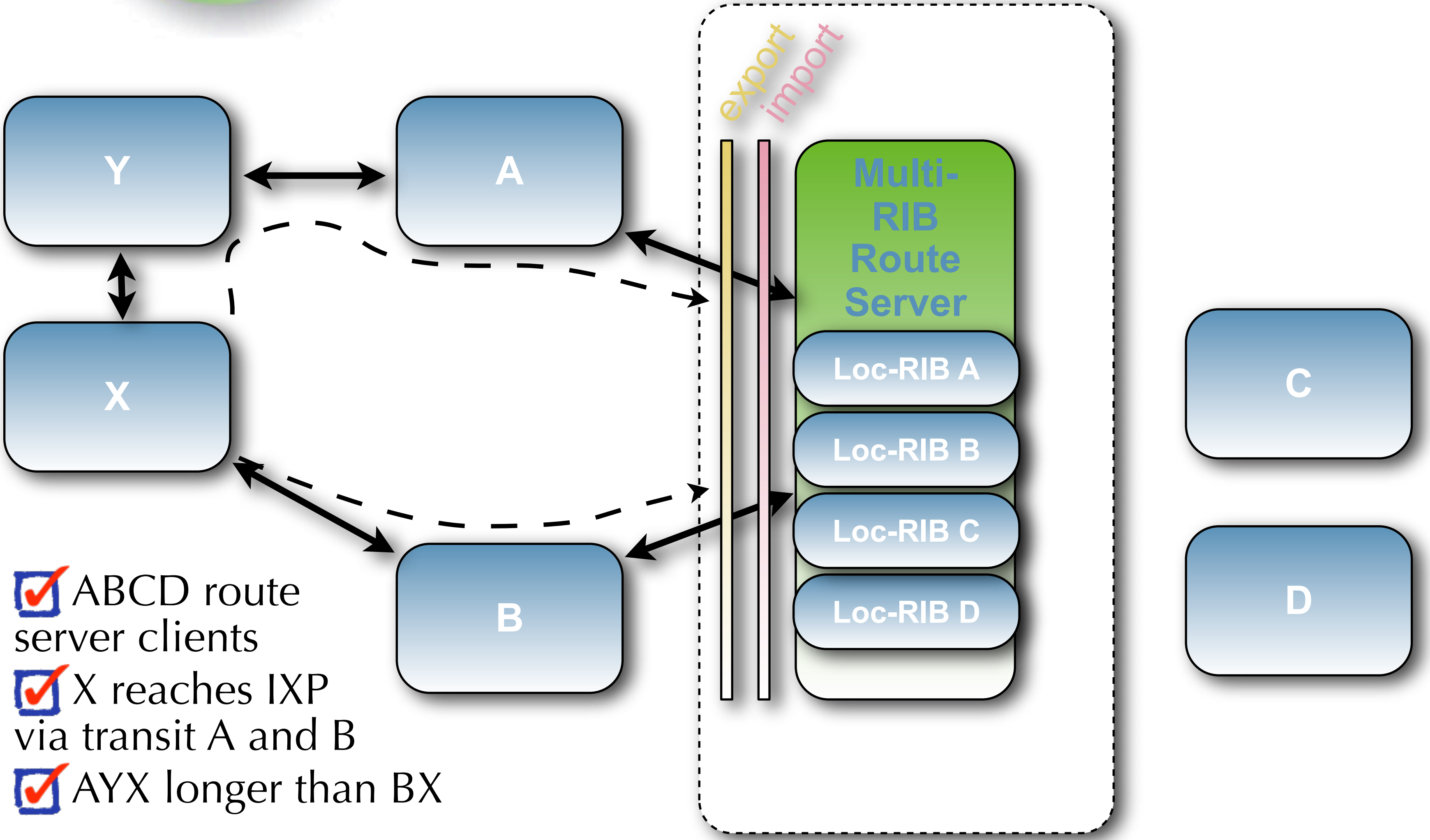
How Per-Client Loc-RIBs Work



☑ ABCD route server clients

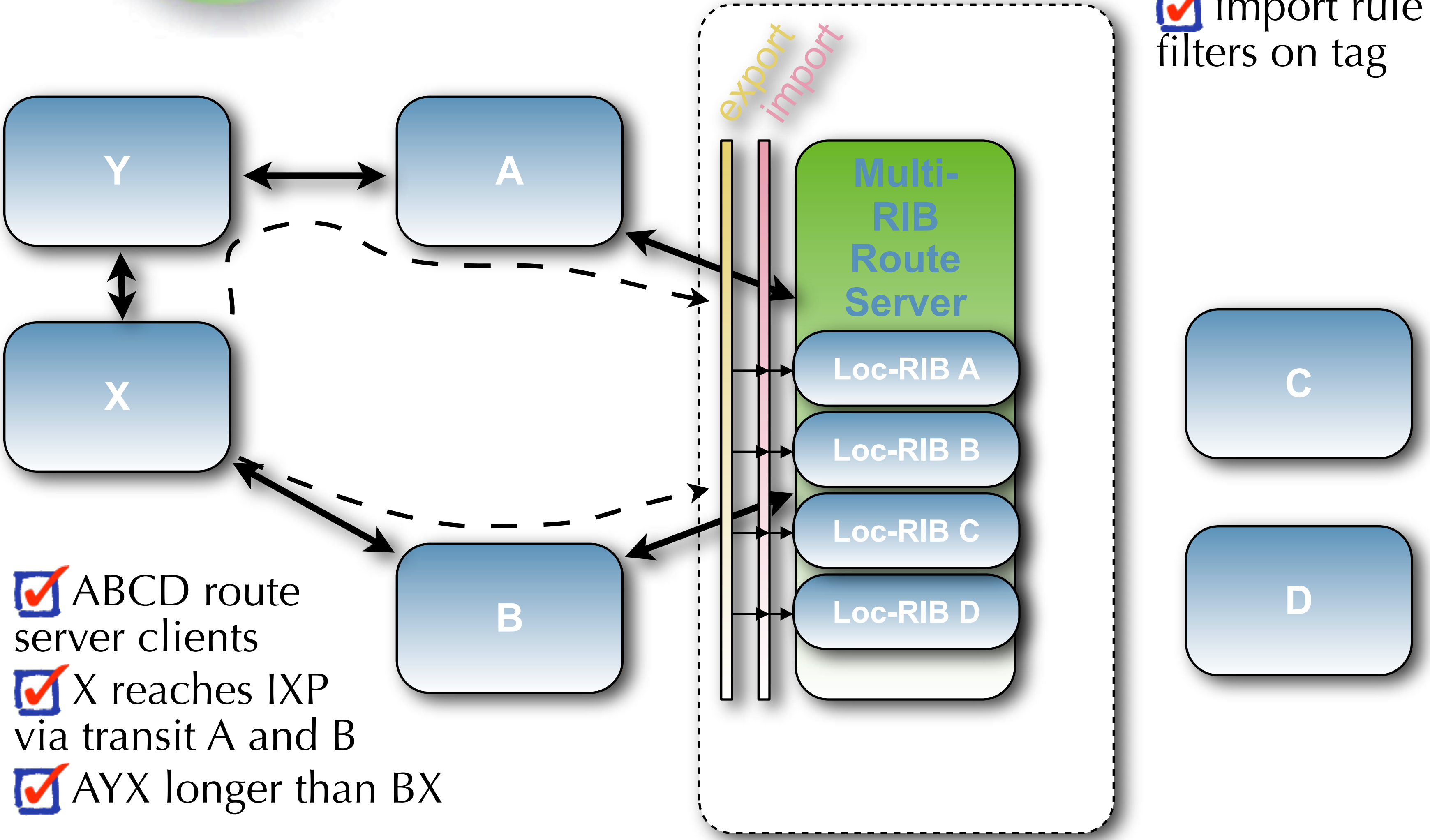
☑ X reaches IXP via transit A and B

How Per-Client Loc-RIBs Work



- ☑ ABCD route server clients
- ☑ X reaches IXP via transit A and B
- ☑ AYX longer than BX

How Per-Client Loc-RIBs Work



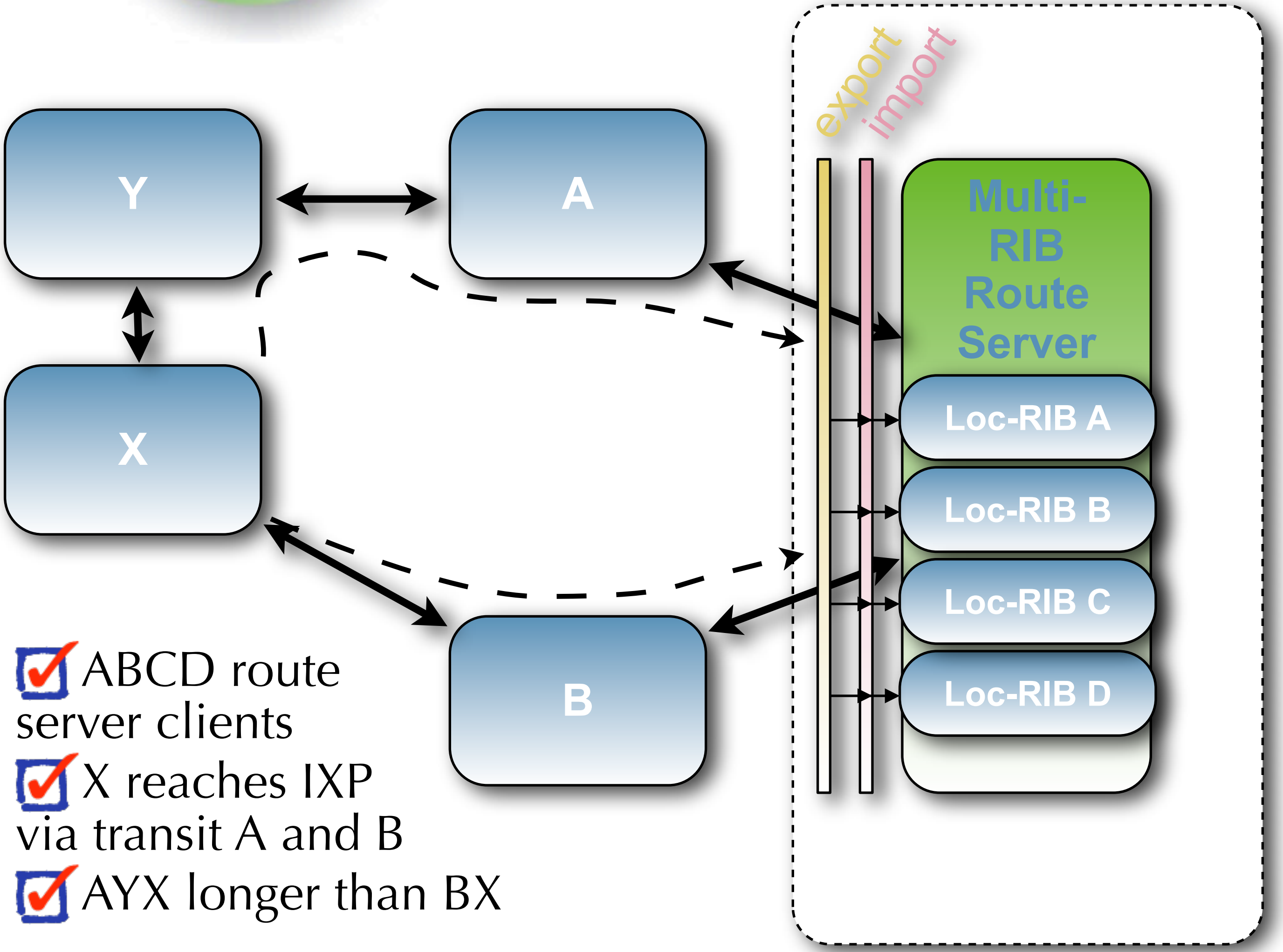
☑ import rule filters on tag

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

How Per-Client Loc-RIBs Work



☑ import rule filters on tag

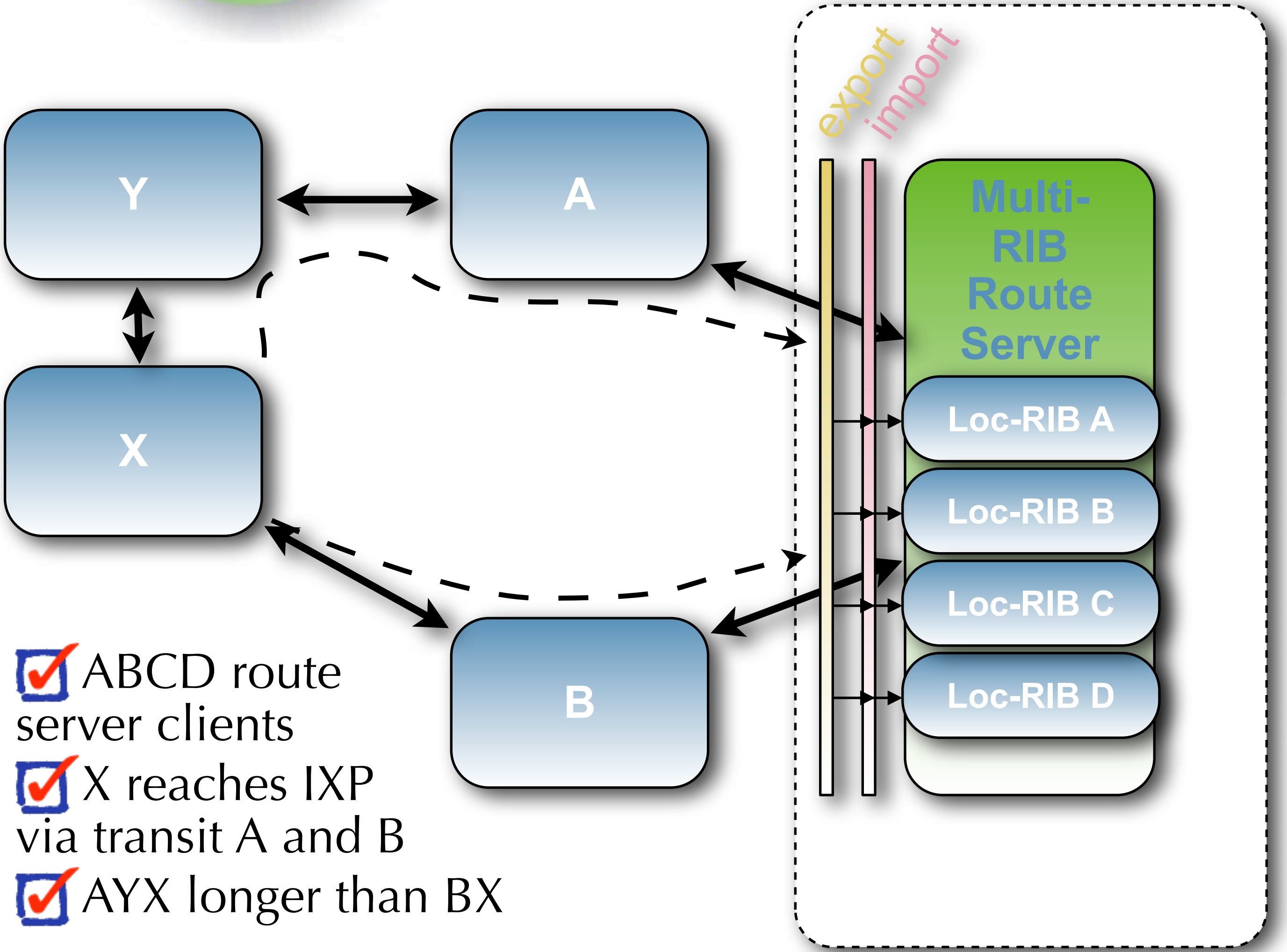
☑ RIB C sees BX and AYX

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

How Per-Client Loc-RIBs Work



☑ import rule filters on tag

☑ RIB C sees BX and AYX

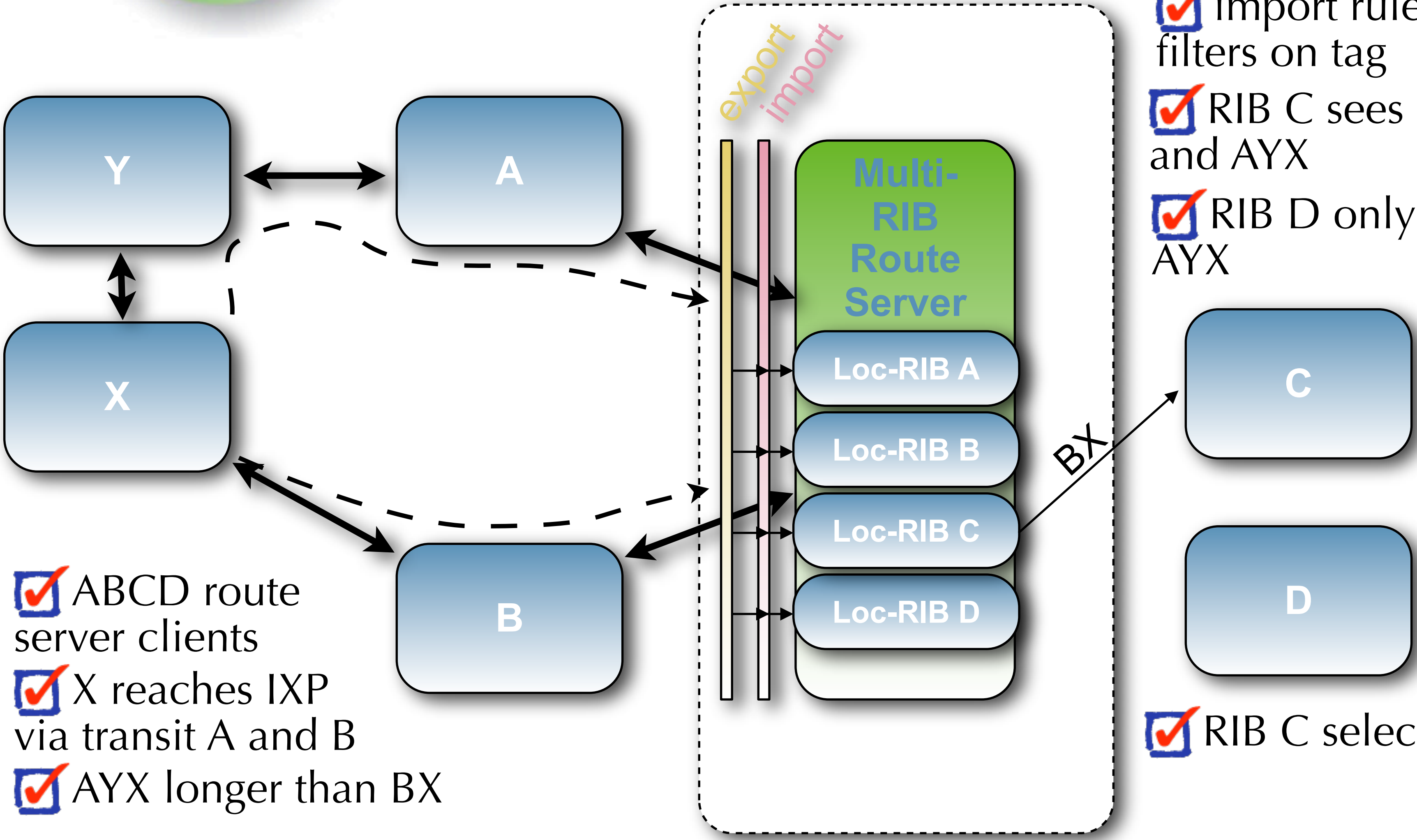
☑ RIB D only sees AYX

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

How Per-Client Loc-RIBs Work



☑ import rule filters on tag

☑ RIB C sees BX and AYX

☑ RIB D only sees AYX

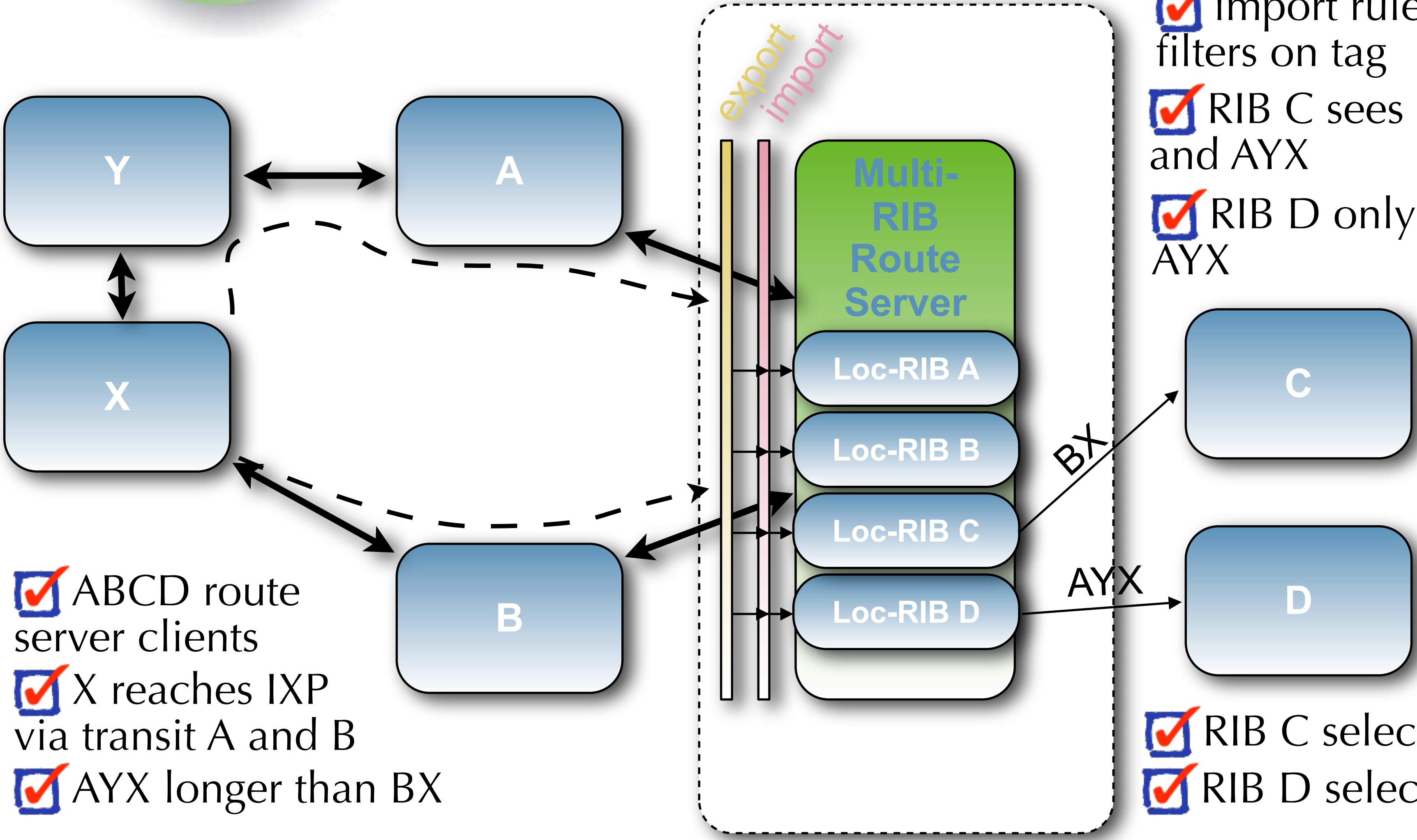
☑ RIB C selects BX

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

How Per-Client Loc-RIBs Work



☑ import rule filters on tag

☑ RIB C sees BX and AYX

☑ RIB D only sees AYX

☑ ABCD route server clients

☑ X reaches IXP via transit A and B

☑ AYX longer than BX

☑ RIB C selects BX

☑ RIB D selects AYX

Ceiling Cat is Watching you Propagate

Problems with Multiple Loc-RIBs

Problems with Multiple Loc-RIBs

- Multiple Loc-RIBs mean:
  - Memory, CPU consumption go from O(M) to O(N x M)
    - N = number of clients
    - M = total number of prefixes

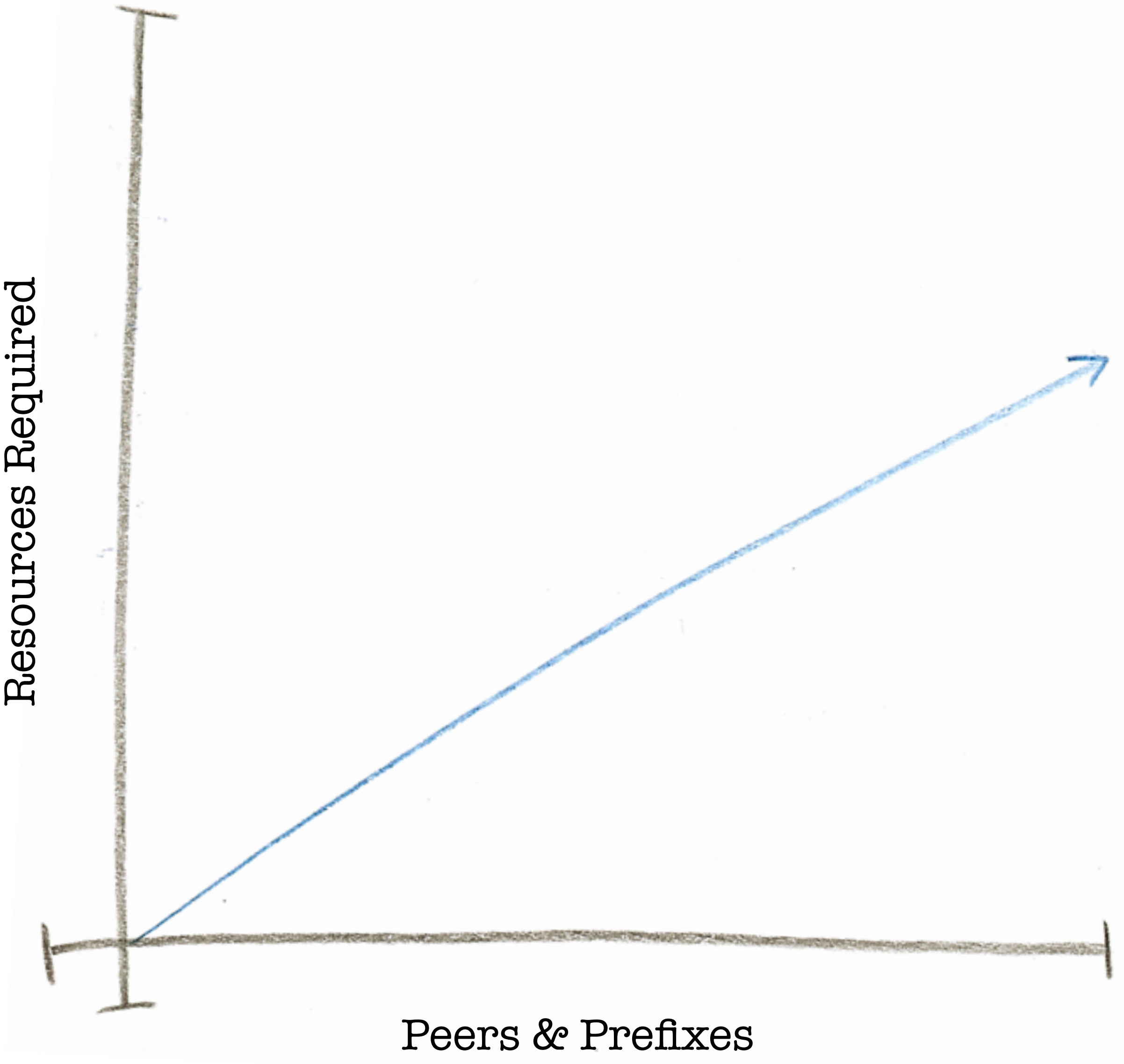
Problems with Multiple Loc-RIBs

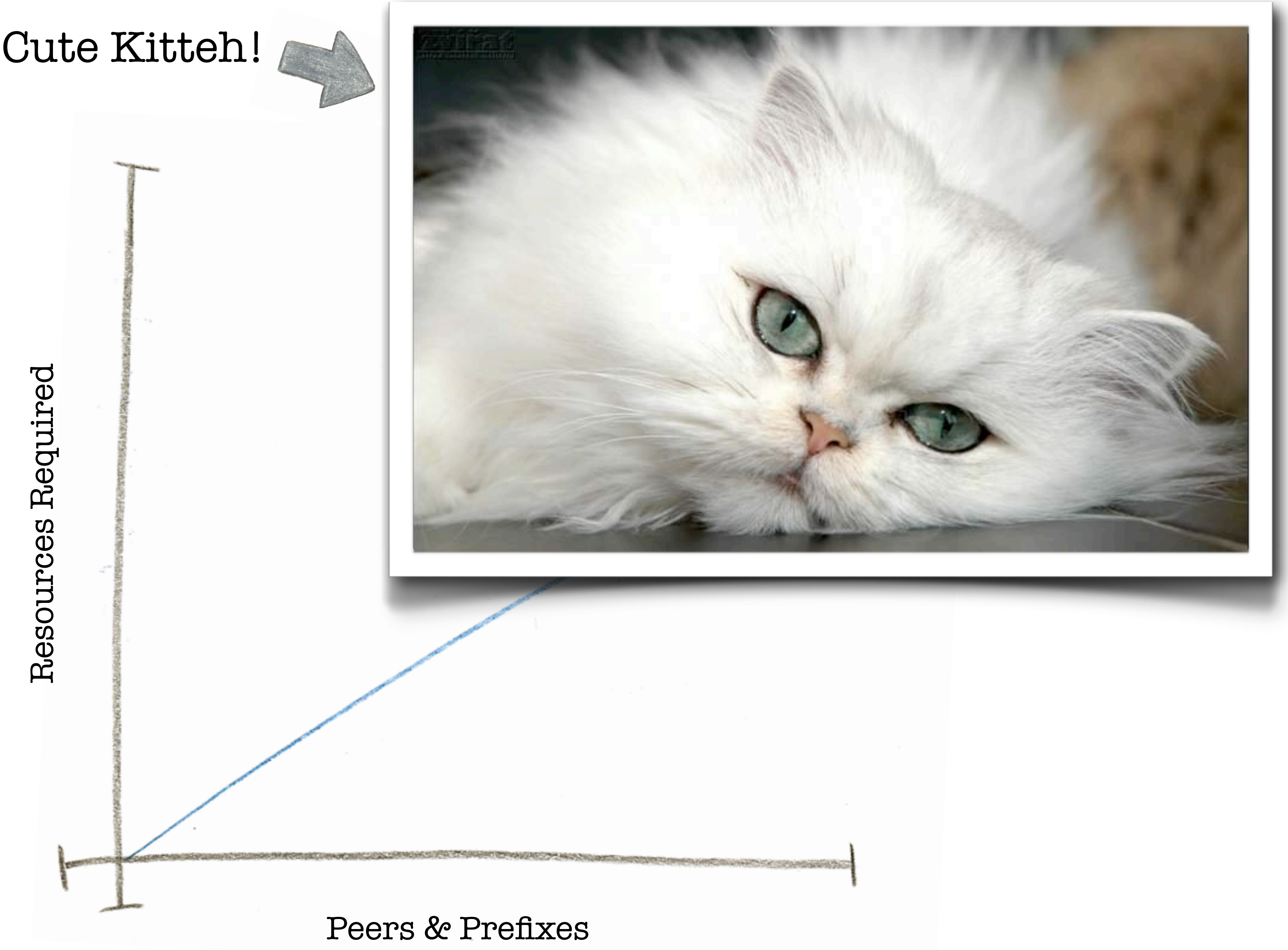
- Multiple Loc-RIBs mean:
    - Memory, CPU consumption go from O(M) to O(N x M)
        - N = number of clients
        - M = total number of prefixes
    - Update processing resources required are:

$$\sum_{1}^{N}(P(n)\bullet(N-1))$$

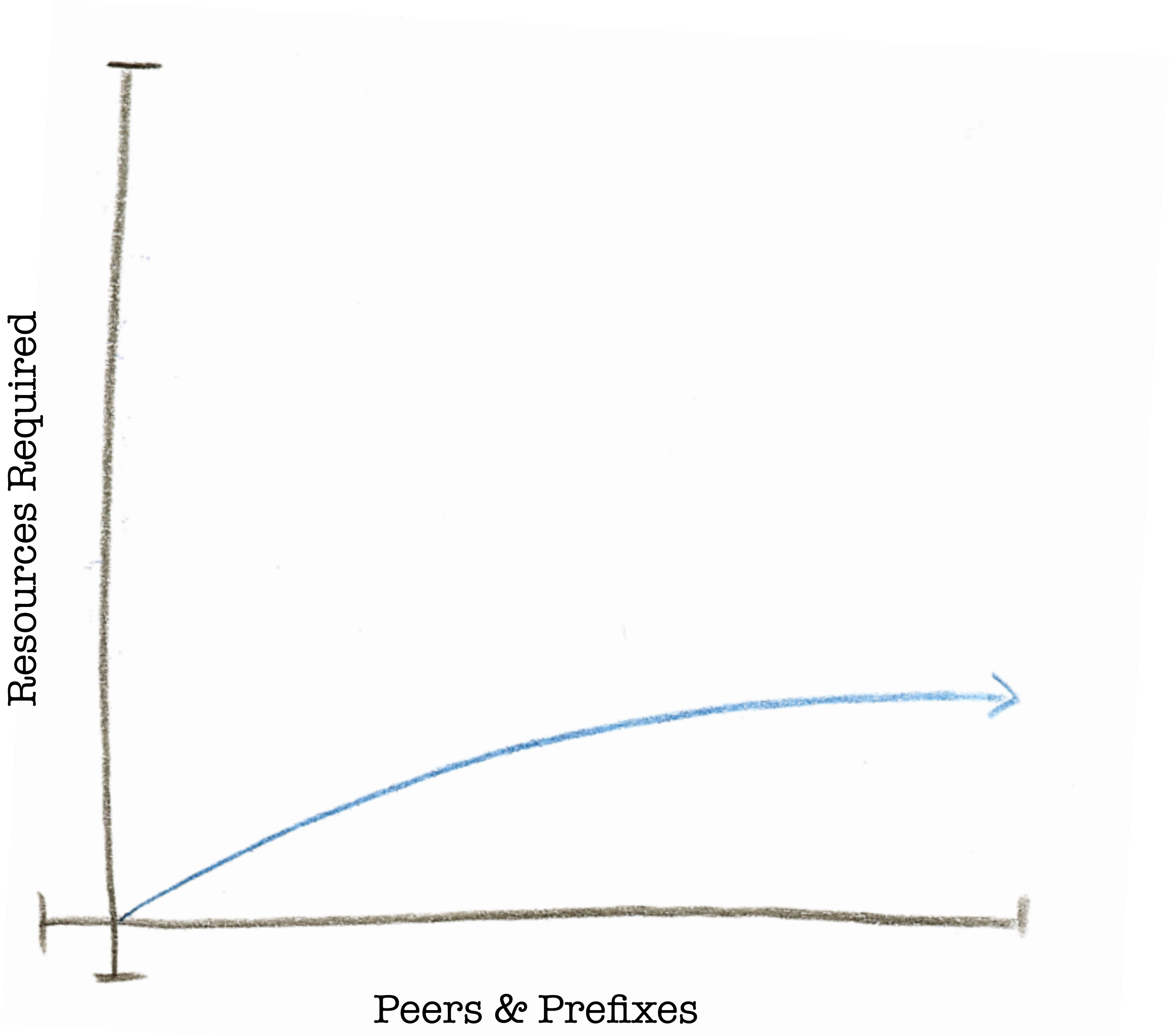
- Where
    - P(n) = the number of prefixes from peer n
    - N = number of peers connected to system
- This scales as $P_{average} * N^2$

Help!

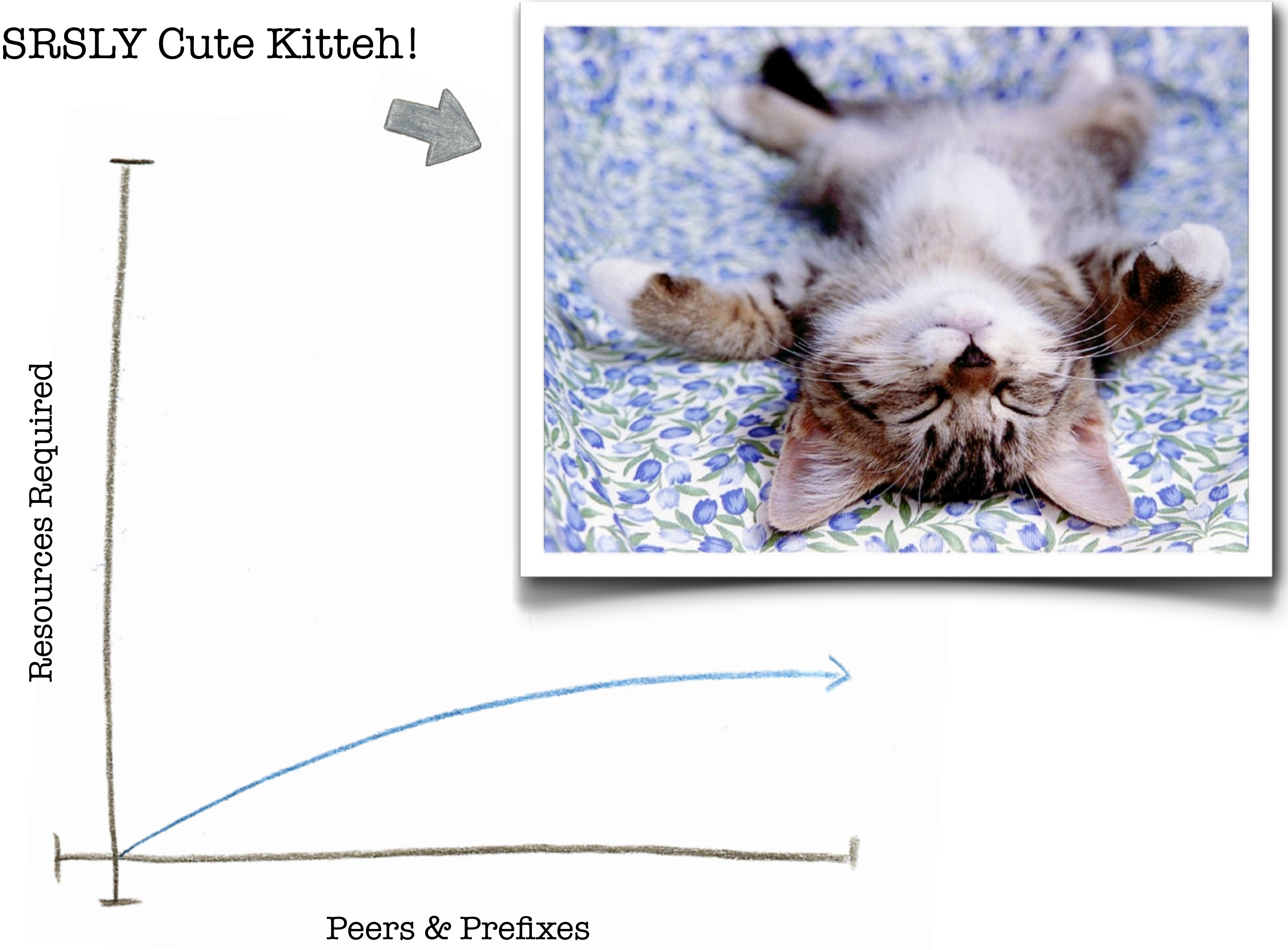

Resources Required
Peers & Prefixes

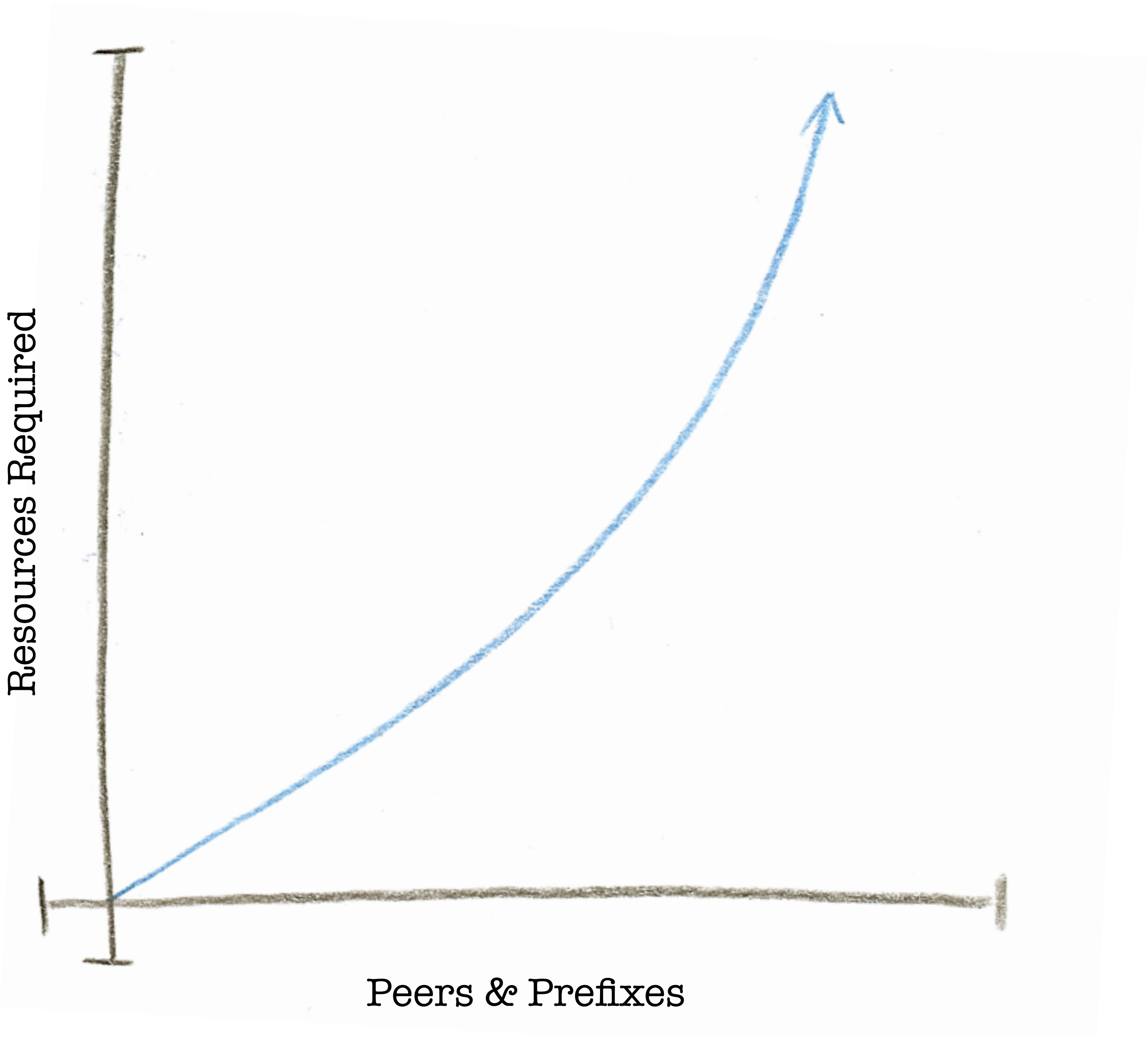


Cute Kitteh!

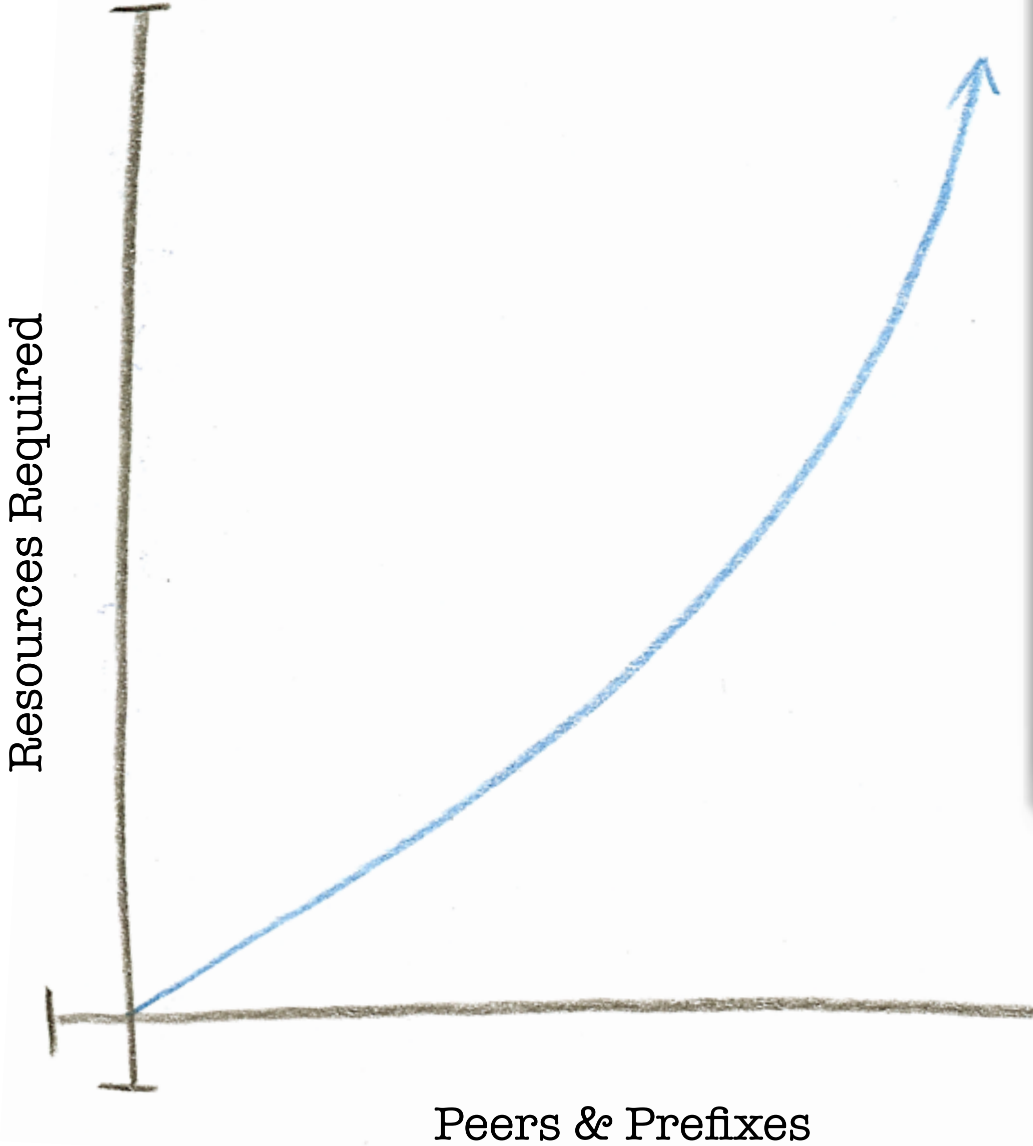

Resources Required
Peers & Prefixes

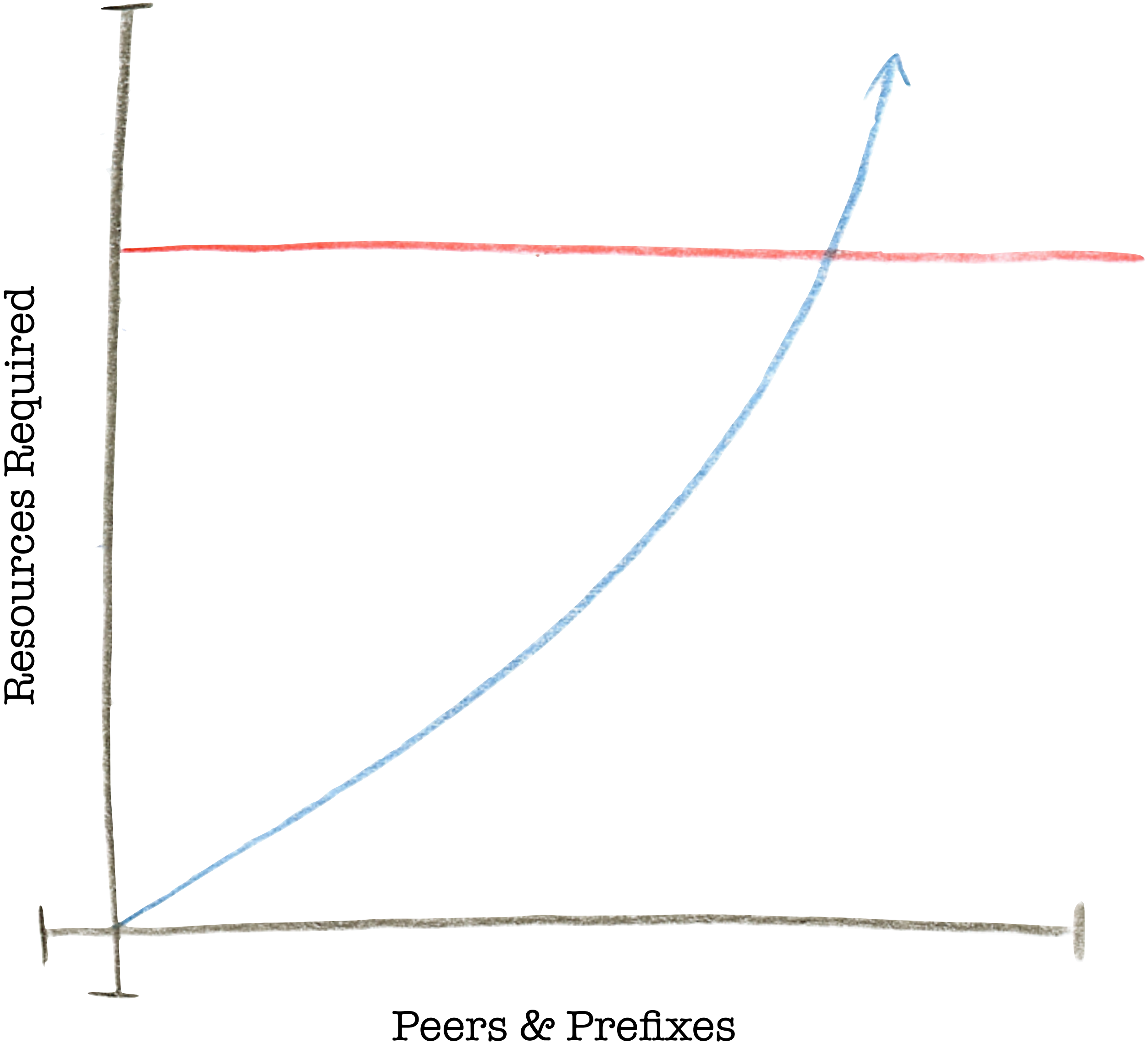
SRSLY Cute Kitteh!

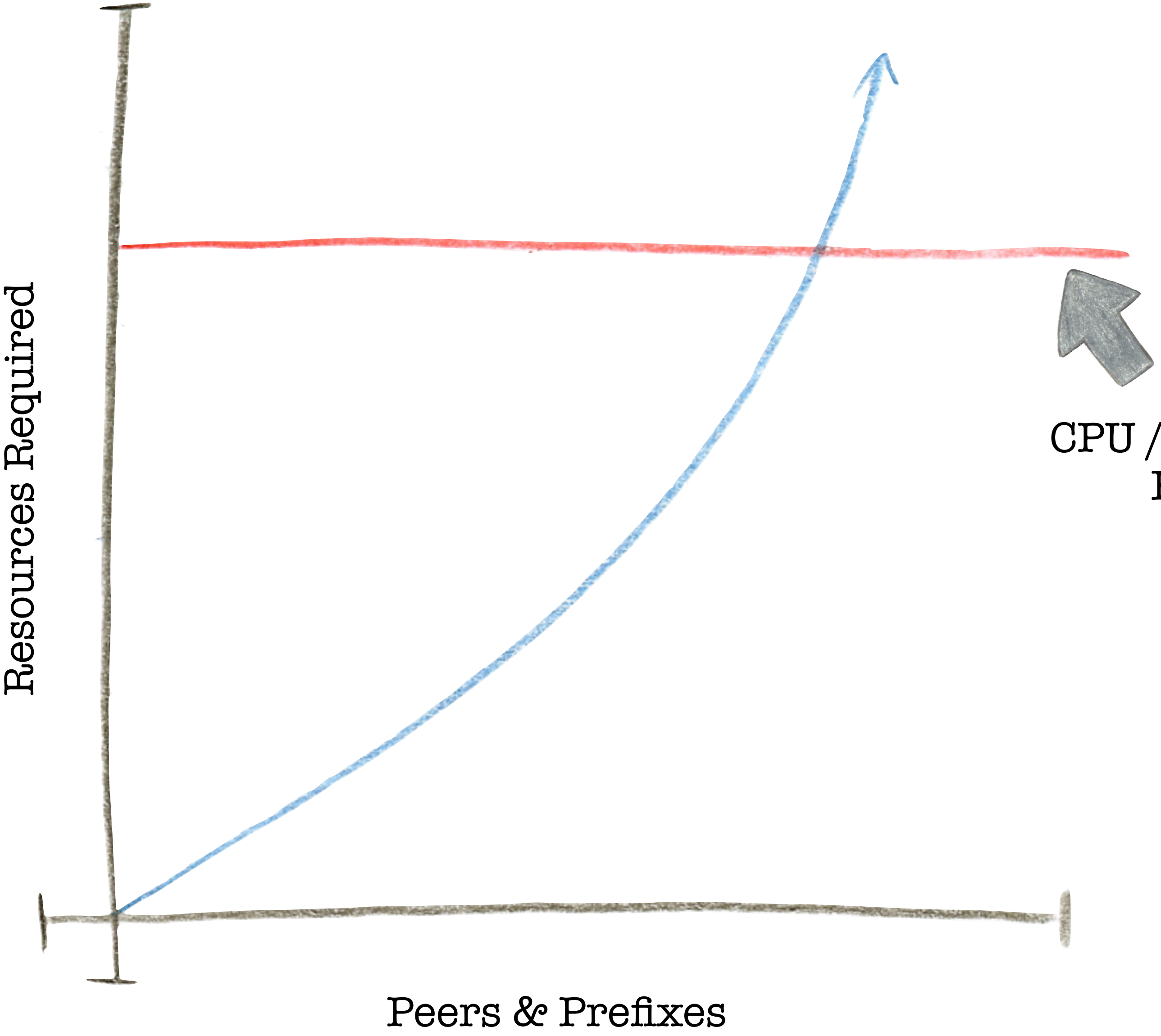



Resources Required
Peers & Prefixes

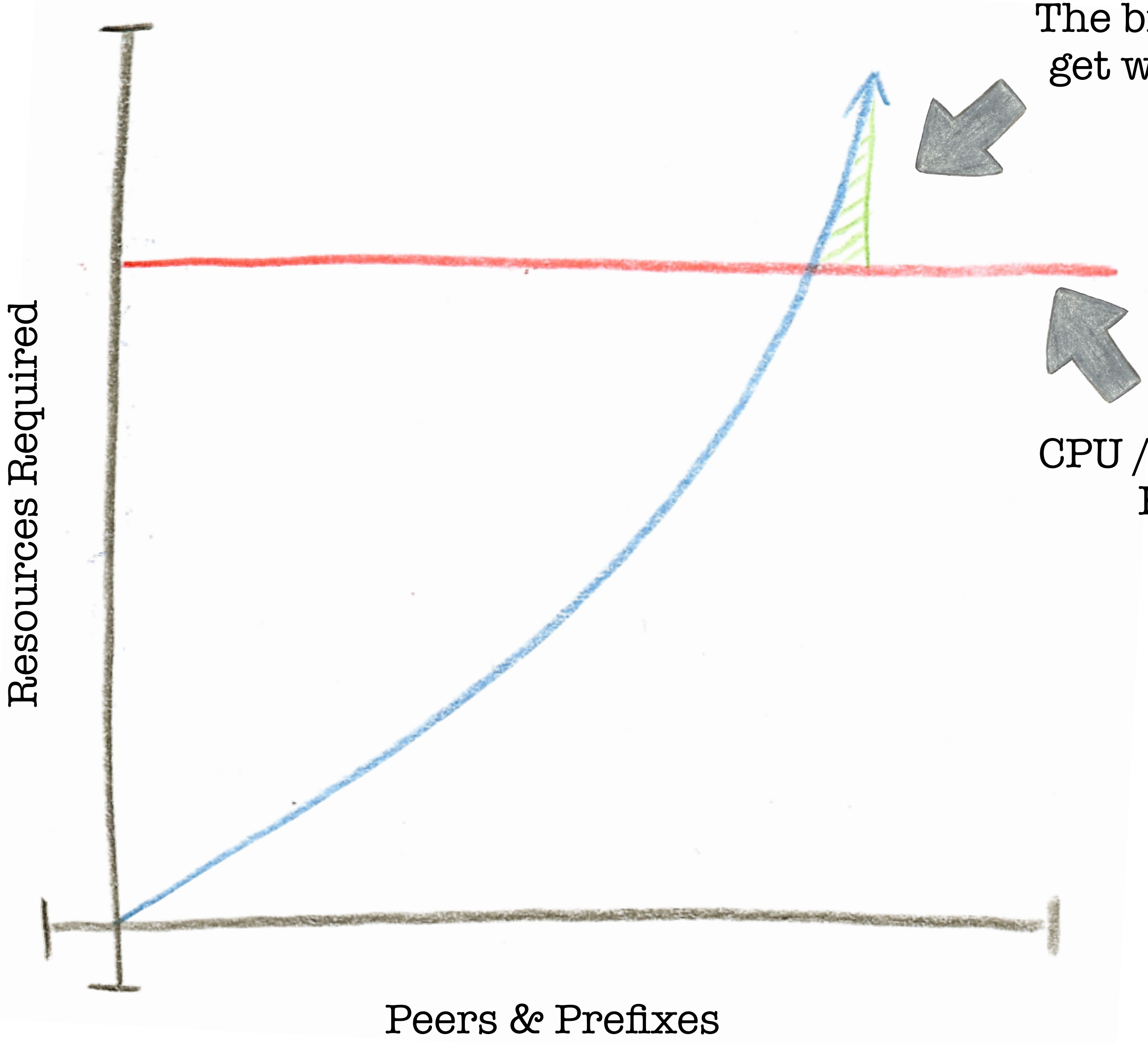
Evil Kitteh!




Resources Required
Peers & Prefixes

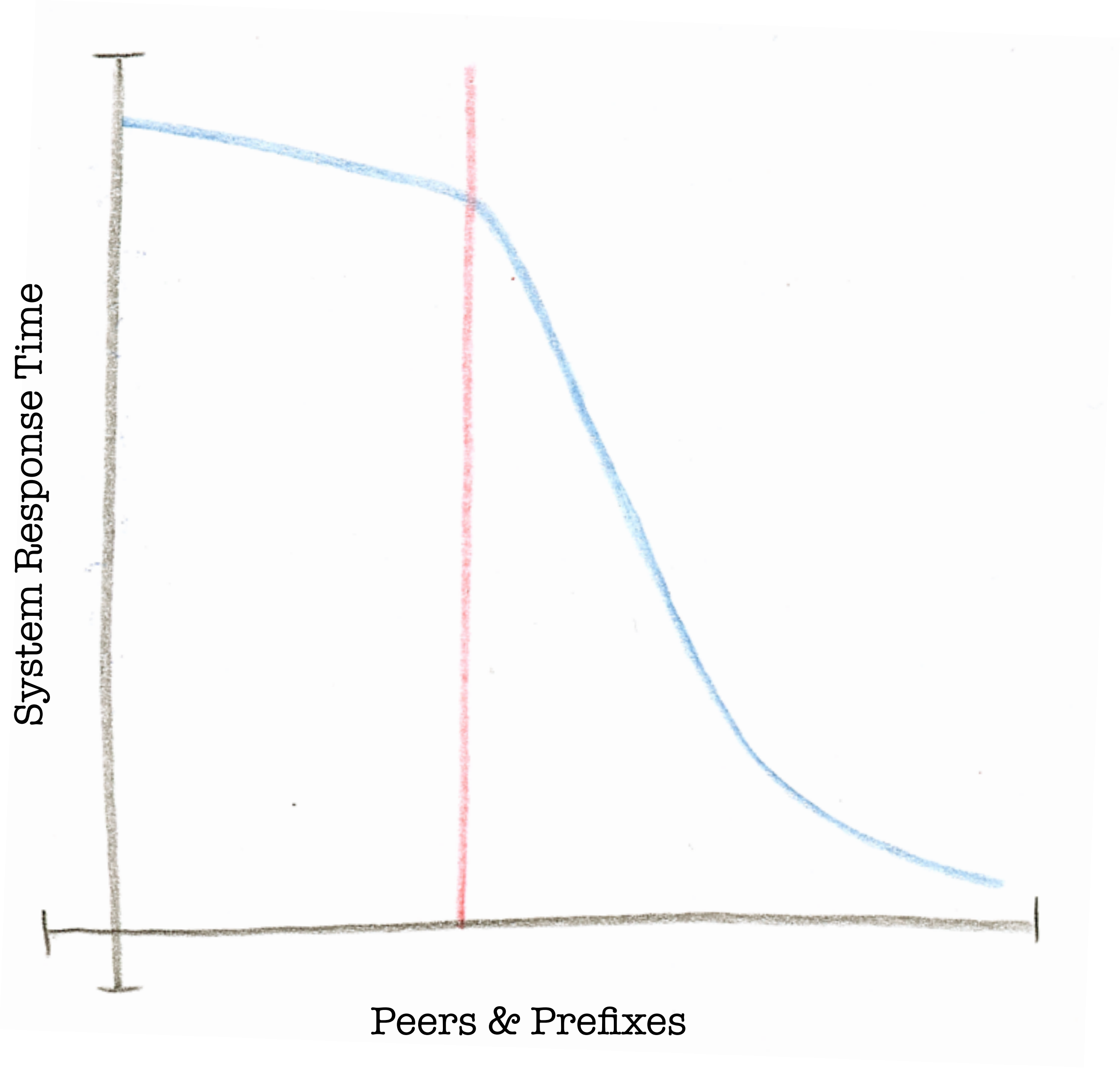


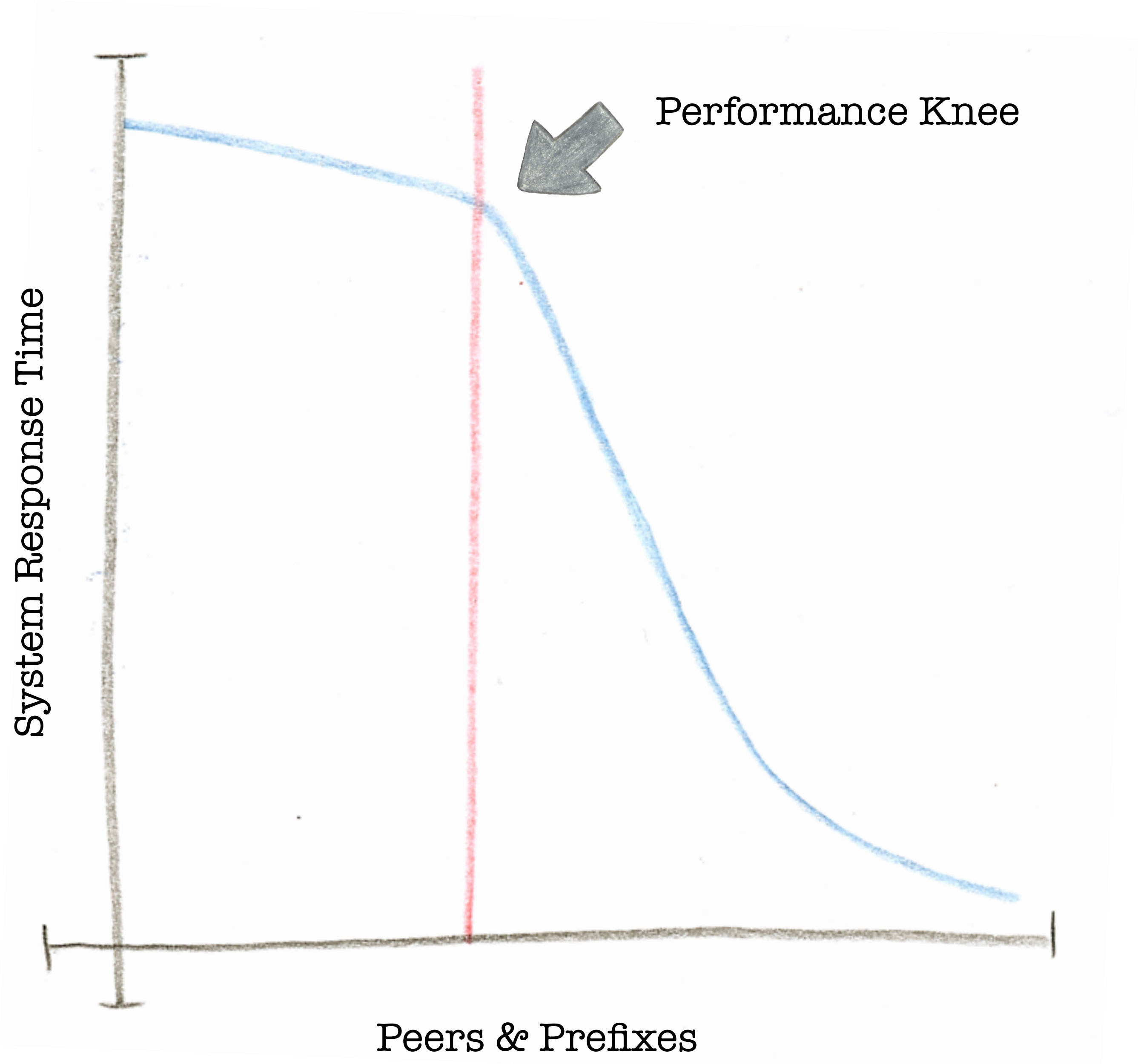
CPU / Memory / Network
Resource Limit

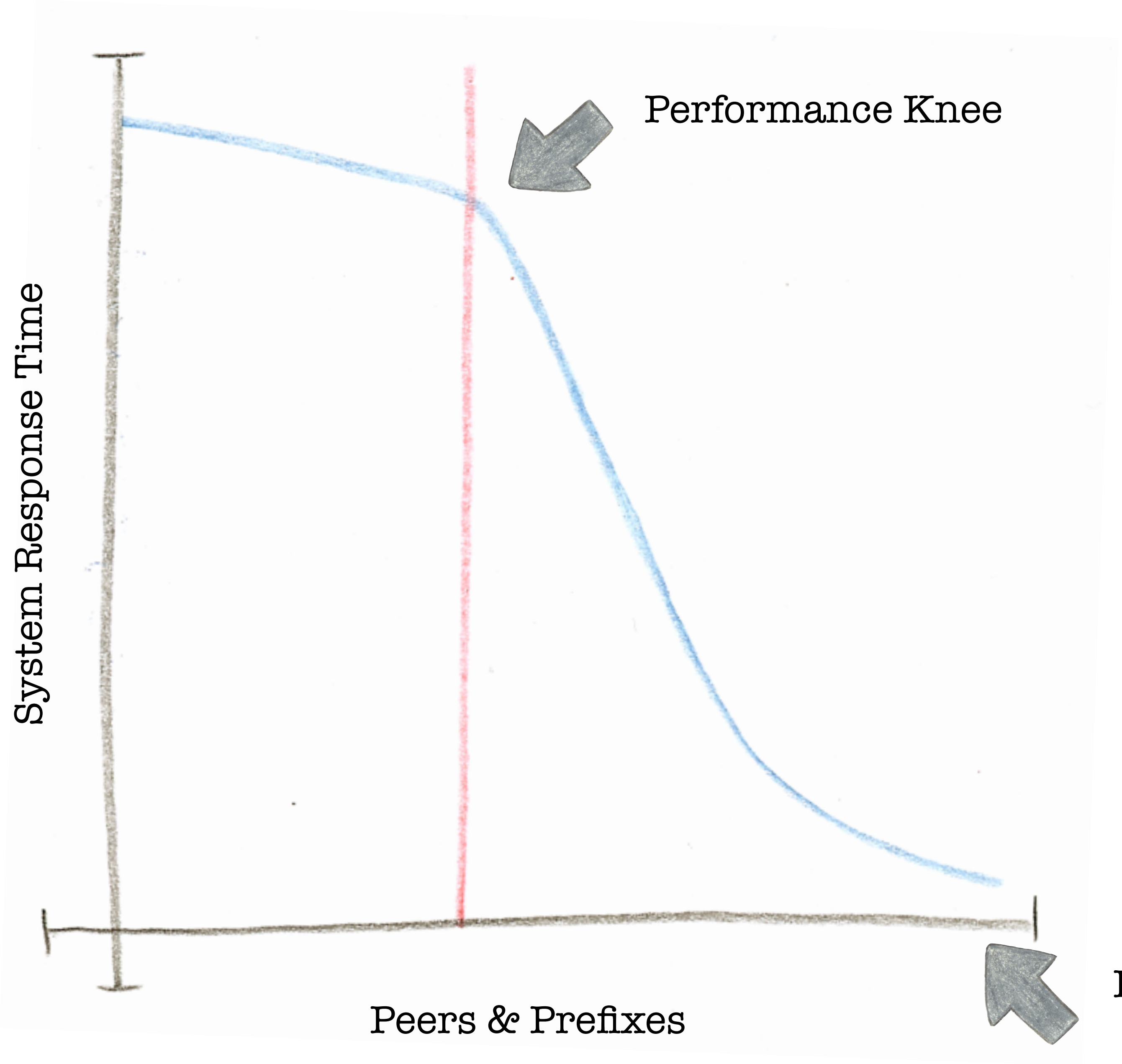


The bit you need to get worried about

CPU / Memory / Network Resource Limit


System Response Time
Peers & Prefixes


Performance Knee
System Response Time
Peers & Prefixes



Performance Knee

Performance Goes to
Hell in a Handcart

Facts and Figures

- A single BGP prefix update
  - might take 10 - 30 bytes on network to send to peer
  - might take 10 - 30µS to process update
- Disclaimer
  - this ignores attributes, path length, cpu speed, and a pile of other highly relevant parameters

Facts and Figures

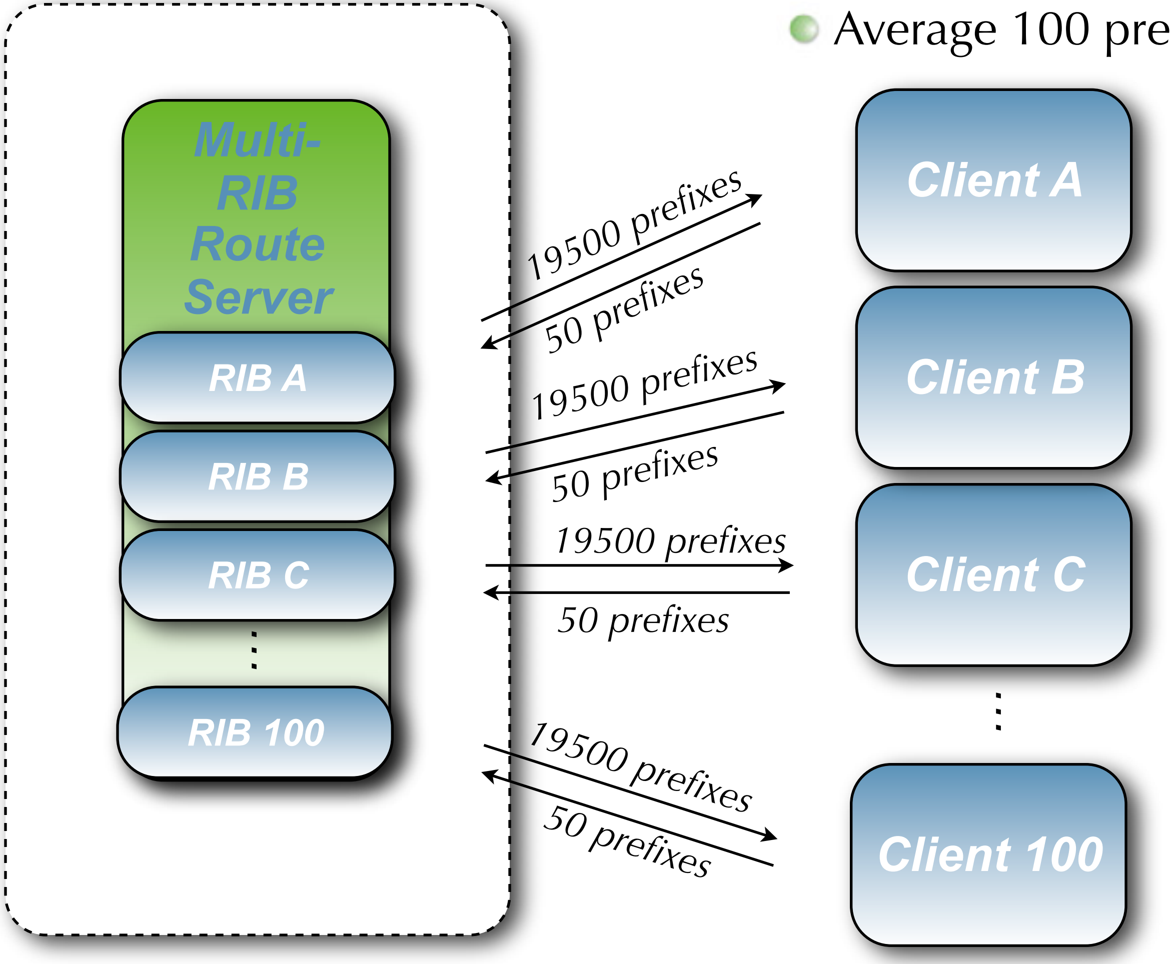
- A single BGP prefix update
  - might take 10 - 30 bytes on network to send to peer
  - might take 10 - 30µS to process update
- Disclaimer
  - this ignores attributes, path length, cpu speed, and a pile of other highly relevant parameters
- Ok, it's hand-waving

Facts and Figures

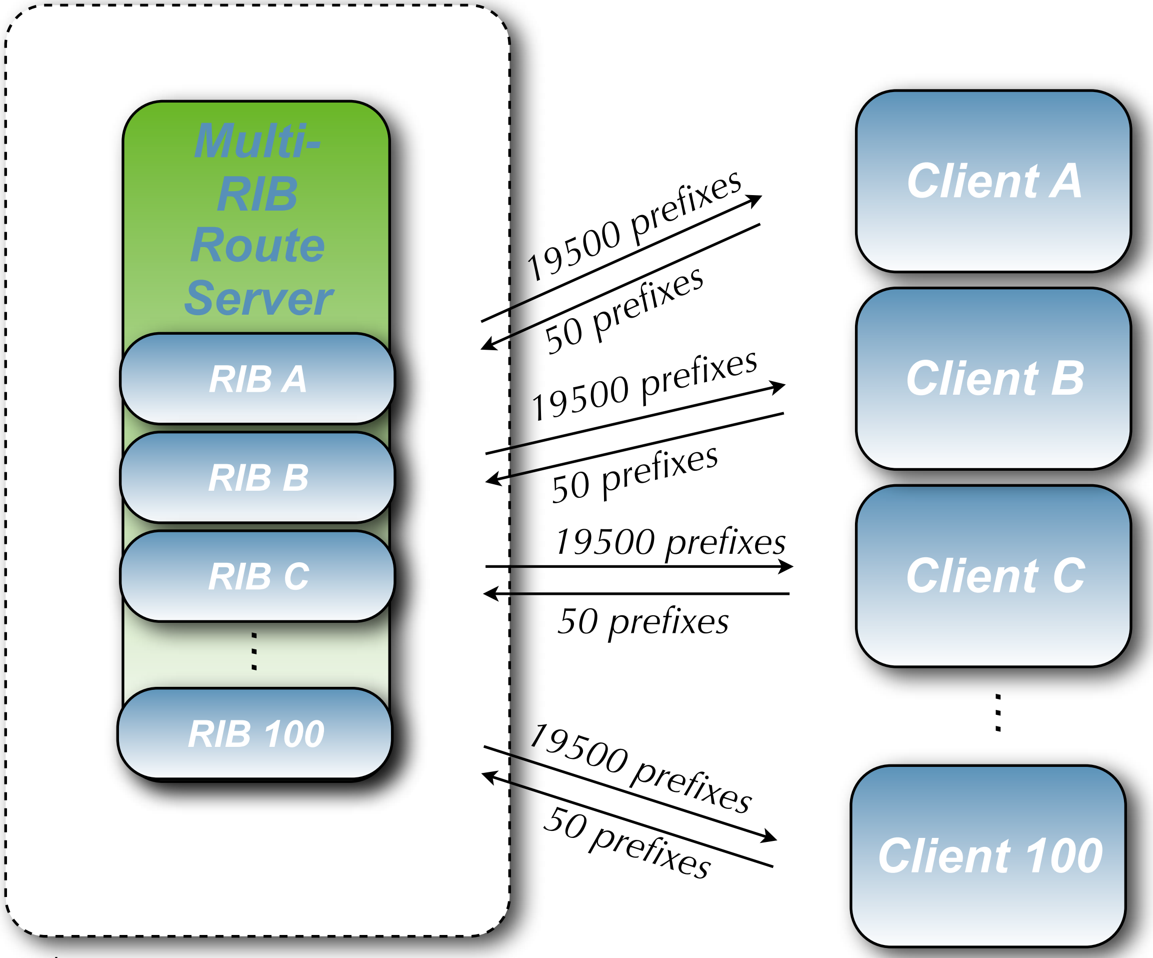
- 200 clients
- Average 100 prefixes each

Facts and Figures

- 200 clients
- Average 100 prefixes each



Facts and Figures



- 200 clients
- Average 100 prefixes each

- RS to Client updates:
  - 19500*200 = 3,900,000
- Client to RS updates:
  - 100 * 200 = 20000 updates
- Total BGP updates: 4,000,000

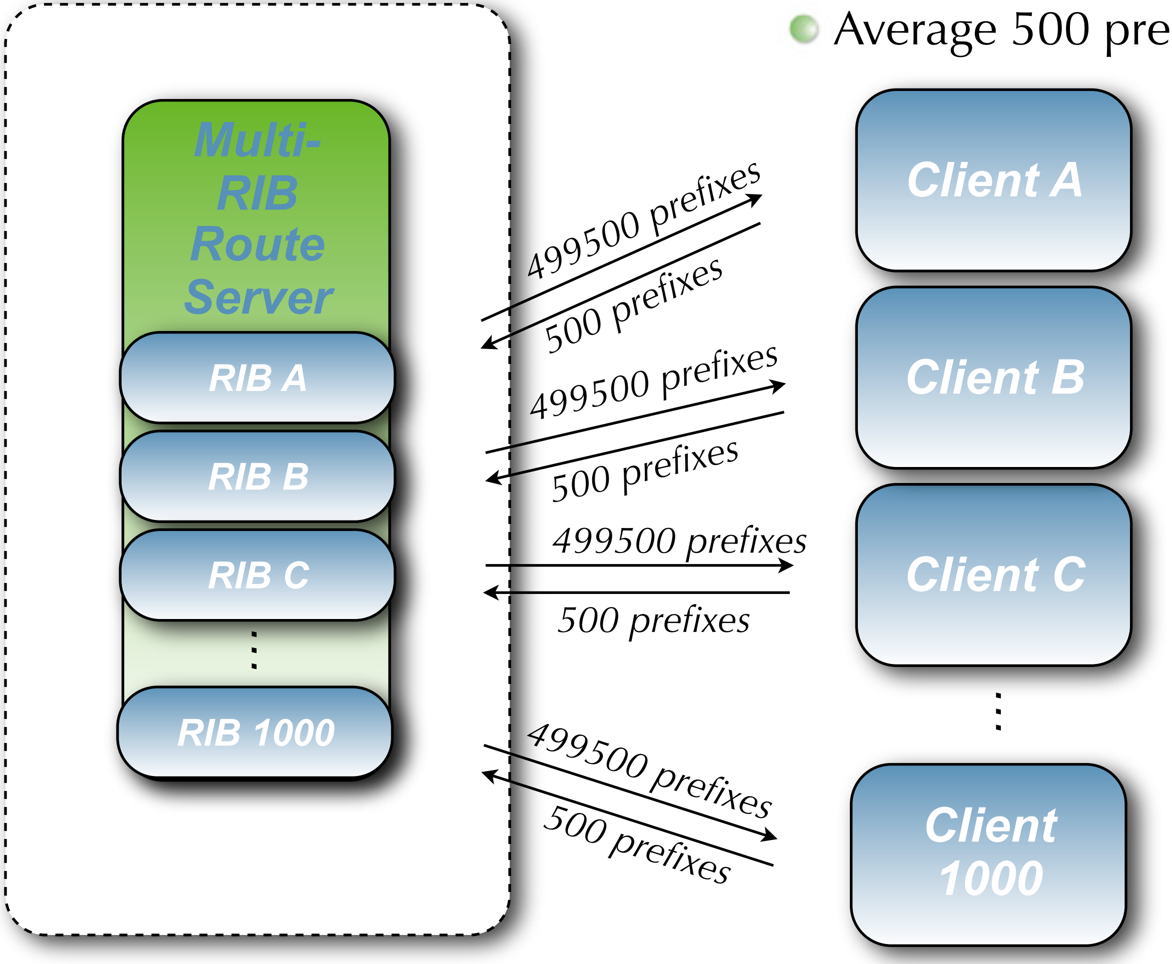
- 40-120M of network traffic
- 40-120s CPU time

Facts and Figures

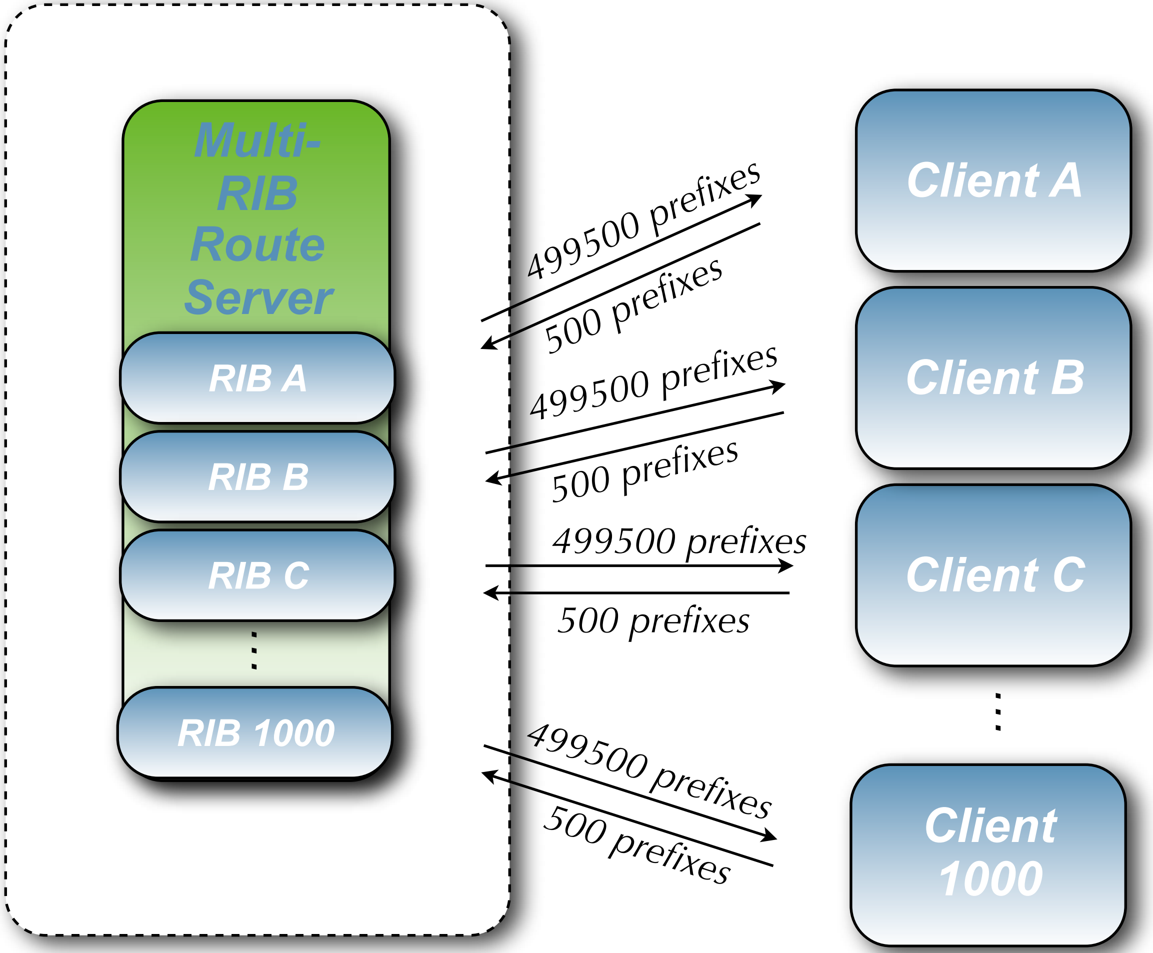
- 1000 clients
- Average 500 prefixes each

Facts and Figures

- 1000 clients
- Average 500 prefixes each



Facts and Figures



- 1000 clients
- Average 500 prefixes each
- RS to Client updates:
  - 499000*1000 = 499.5m
- Client to RS updates:
  - 1000 * 500 = 500k updates
- Total BGP updates: 500m
- 5-15G of network traffic
- 85 - 250m CPU time

How Do We Fix This?

- Super-linear scaling causes inherent breakage
  - Moving away from one Loc-RIB per client model is critical
  - Right now, this isn’t the primary cause of IXP breakage
- Three primary models to escape this limitation
  - Collapse multiple Loc-RIBs in memory into single gargantuan Loc-RIB
    - less memory, less CPU
    - “You can run but you can’t hide”
  - Use prior knowledge
    - “Web based peering”
    - Disable unique Loc-RIB on per client basis
  - BGP `ADD_PATH` Capability
    - Published as ID: `draft-walton-bgp-add-paths`
    - Moves BGP Best Path Selection to client, so filtering can be performed without selecting