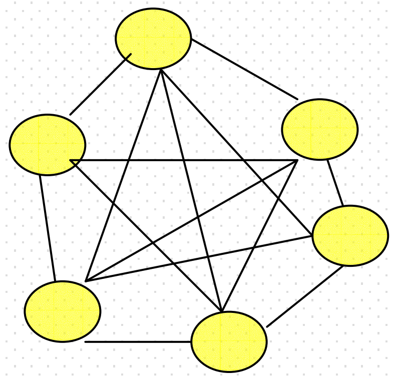
An Analysis of route reflector performance in I-BGP

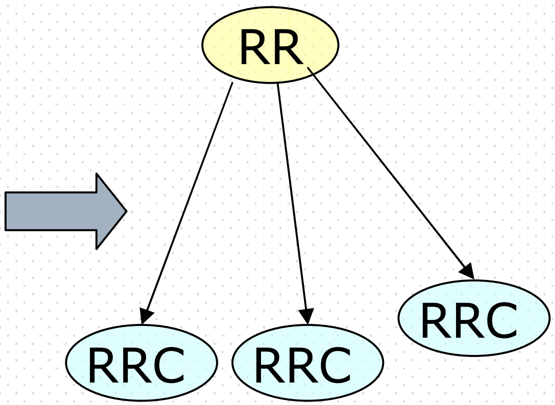
Kengo NAGAHASHI
The University of Tokyo/WIDE Project
kenken@wide.ad.jp

Background(1)

- I-BGP
  - Requires synchronization with all I-BGP routers
    - Full mesh
  - ⇨ Lack of scalability



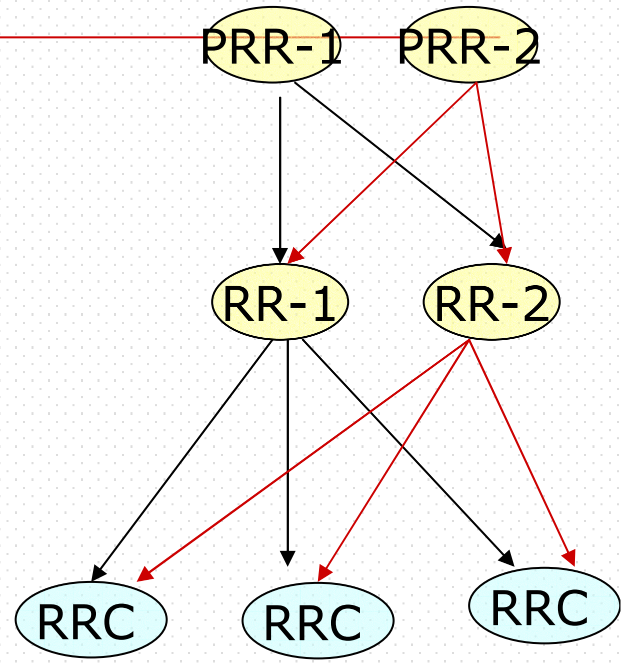
I-BGP fullmesh



Route reflector

- ⇨ Introduction of Route Reflector(RR)

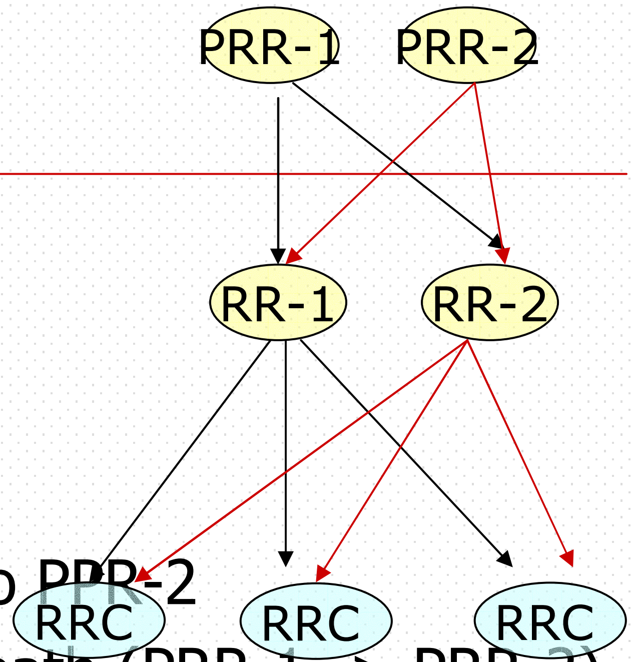
Background(2)



- What if RR is outage?
  - RRCs lost connectivity
  - single point of failure
  - ISP requires 24 hours x 365
  - Requirement for redundancy

- Introduction of Backup RR
  - RRC establishes BGP peer with both RR-1/RR-2
  - RRC receives an exact routing information both from RR-1,RR2
  - **Hierarchal Route Reflector Model**

Problem Statement



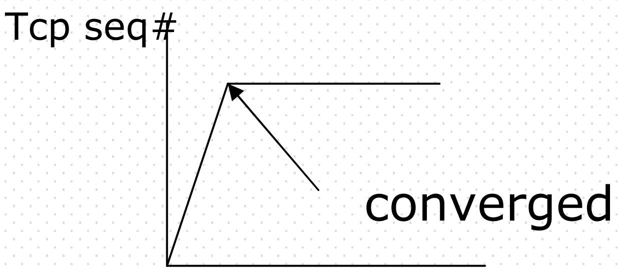
- Problem in this model:
  - Possible case:
    1. PPR-1 (best path) is down
    2. Switch over the best path to PPR-2
    3. RR-1 recalculates the best path (PRR-1 -> PRR-2)
    4. Then , PR-1 sends BGP updates to all RRC despite of all exact routing information (PRR-1 = PRR-2)
- It's due to hop by hop BGP protocol architecture

Motivation

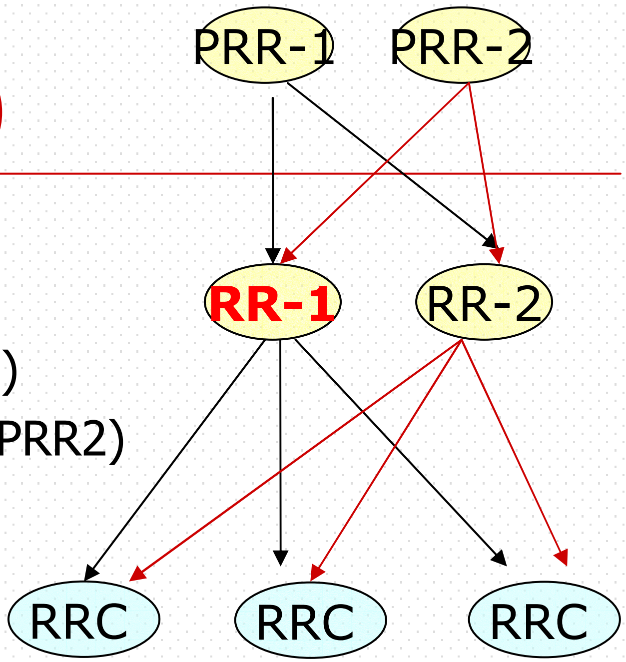
- Is this redundant route reflector architecture truly scalable?
  - How much RRCs can RR accommodate?
    - 10, 100, 1000?
  - What is the main elements which affect a performance of scalability?
    - # of routing information , e.g. fullroute (over 150,000)
    - BGP attribute?
    - Router implementation?

Measurement Strategy(1)

- How can we figure out "scalable or not"?
  - ⇨ Definition of "scalable": convergence of RR/RRCs even if # of RRCs is increased
- How can we measure "convergence"?
  - Convergence: all BGP routing table has been exchanged between RR/RRCs
  - Measurement of TCP sequence:



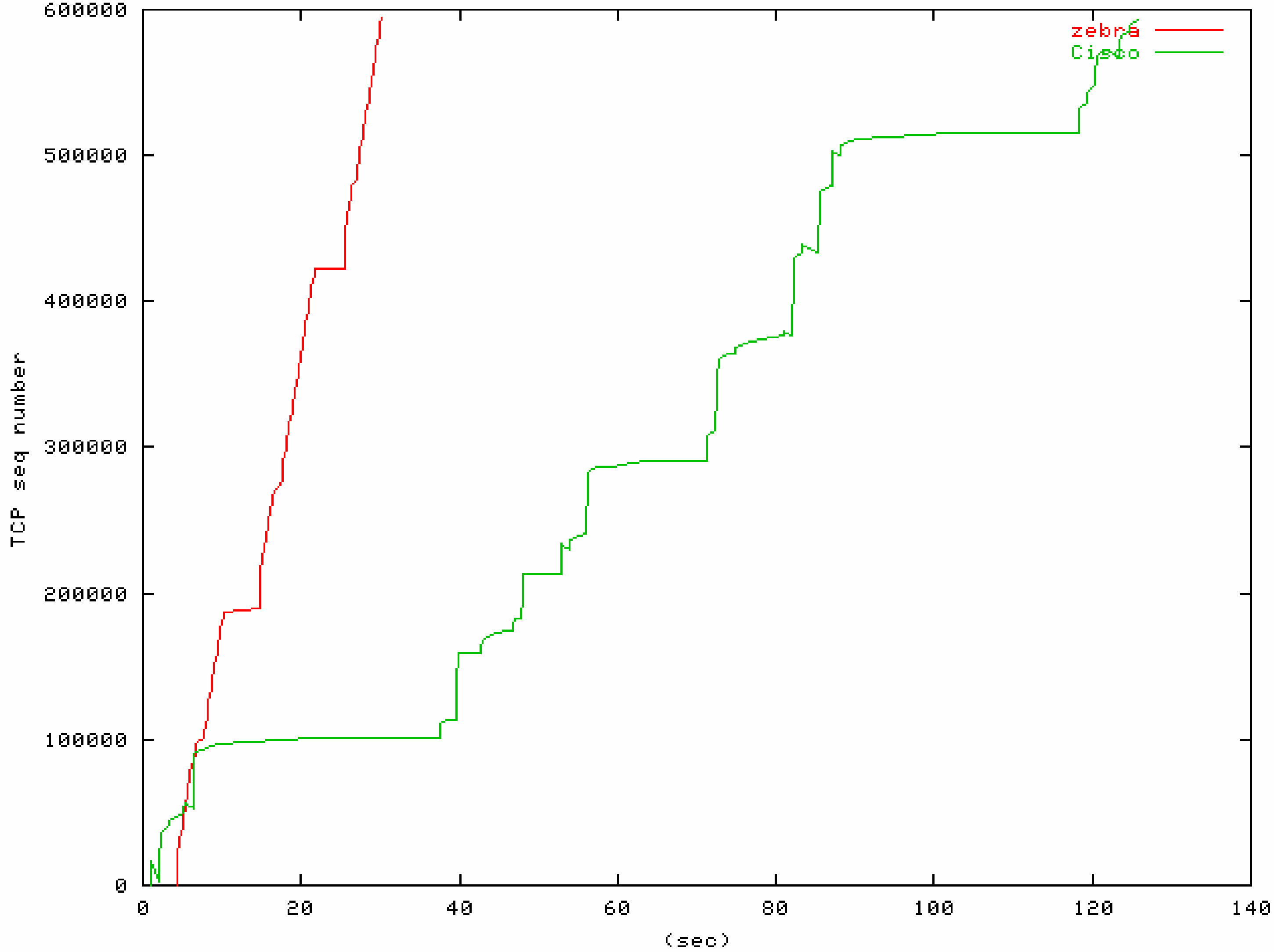
Measurement Strategy(2)



- Triggered event:
  1. Terminate BGP session (PRR-1<->RR-1)
  2. Best path has been changed (PRR-1-> PRR2)
  3. RR-1 recalculates best path
  4. RR-1 sends updates to each RRCs
  5. Measure TCP sequence # in RRC

- Parameters:
  1. BGP table ⇒ full route (146,955prefix/32000 attributes)
  2. RRClient ⇒ 1,30,60,170 RRCs (starbed)
  3. Implementatin ⇒ zebra (FreeBSD4.10,memory 512MB)
     Cisco(IOS12.2(24a)) 256MB FE as RR-1

Measurement Result: 1 RRC(1)



Measurement Result: 1 RRC(2)

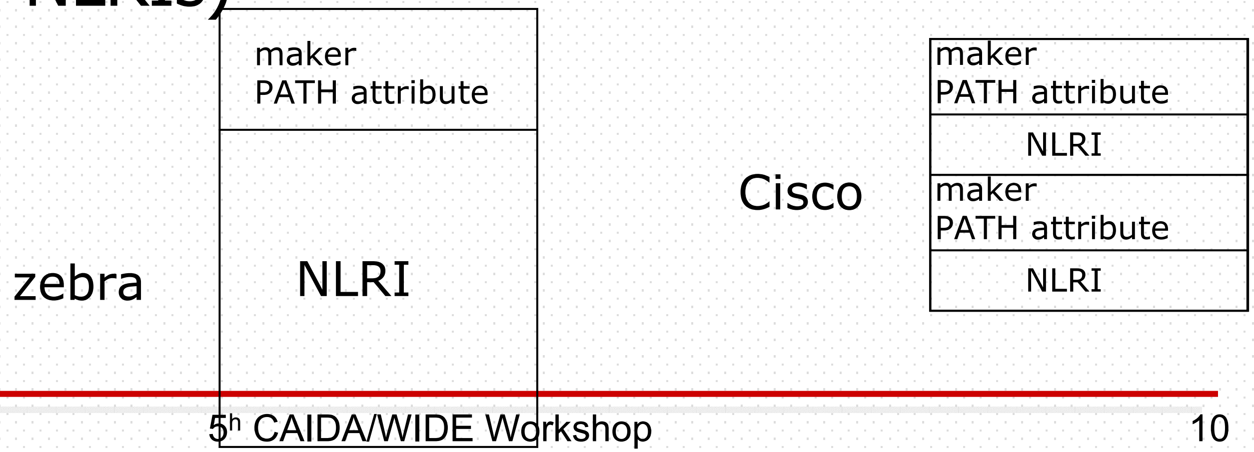
- Convergence of zebra is much faster than Cisco
- Convergence time:
  Cisco 125sec
  zebra 25sec
- MSS problem?

```
cisco>show ip bgp nei | include max data
    Datagrams (max data segment is 1460 bytes):
```

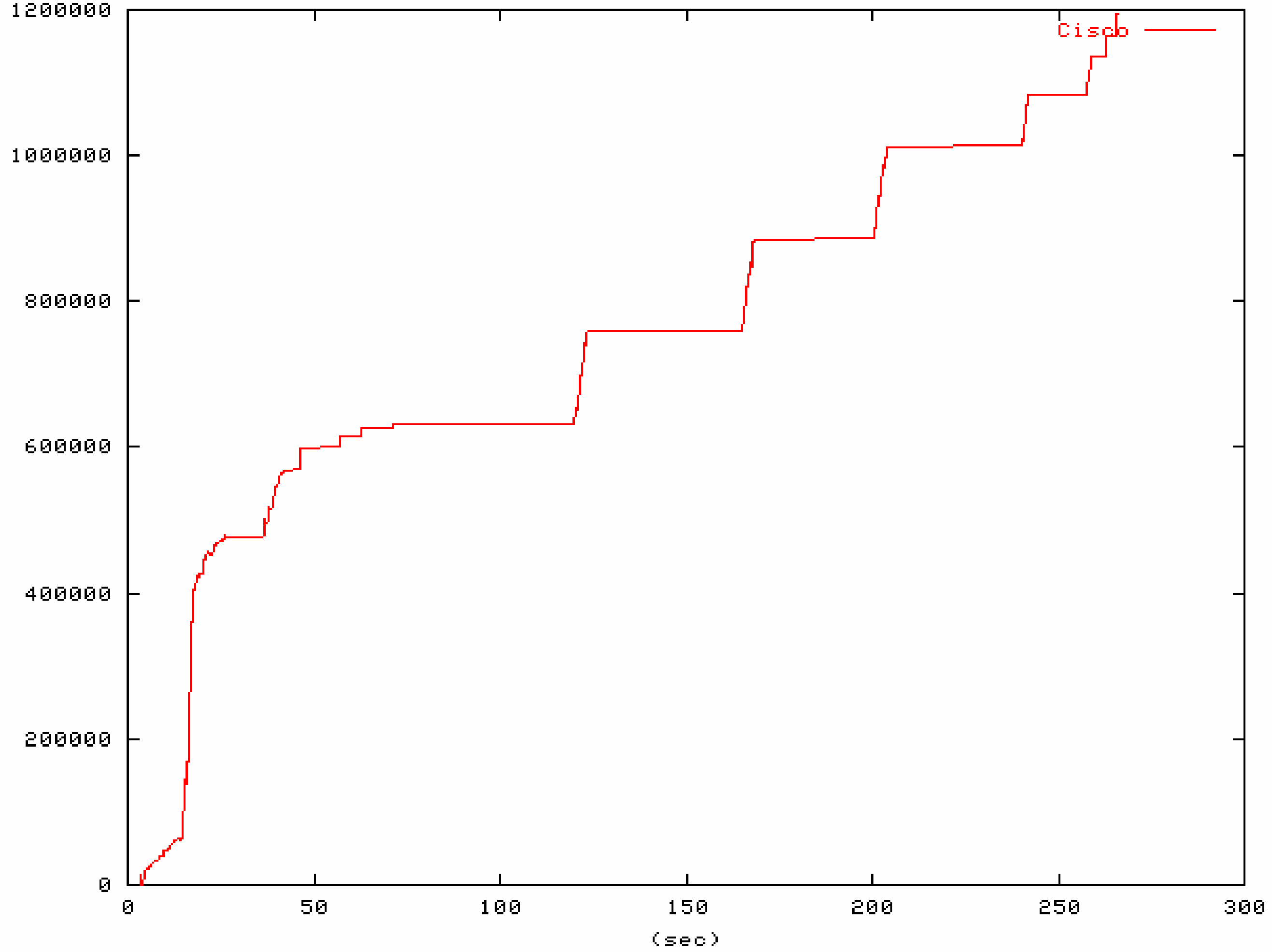
NO

Measurement Result: 1 RRC(3)

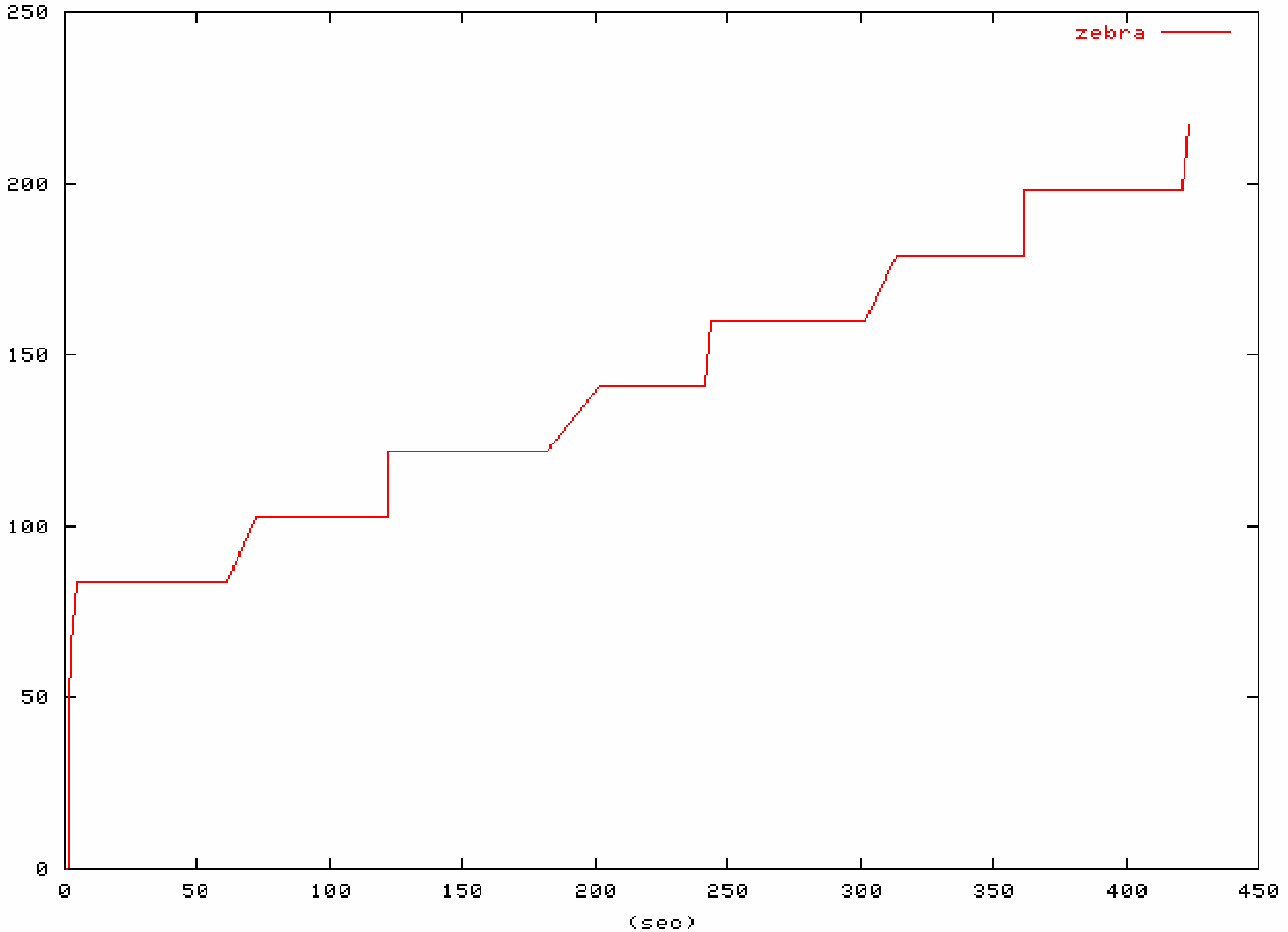
- BGP update packing is different
  - Zebra: packing NLRIs as much as possible in a single BGP Update packet (4096bytes,1000NLRIs)
  - Cisco: chunk 255bytes automatically and if an attribute is same , piggy back one packet (at most 50 NLRIs)



Measurement Result: 60 RRCs (Cisco)



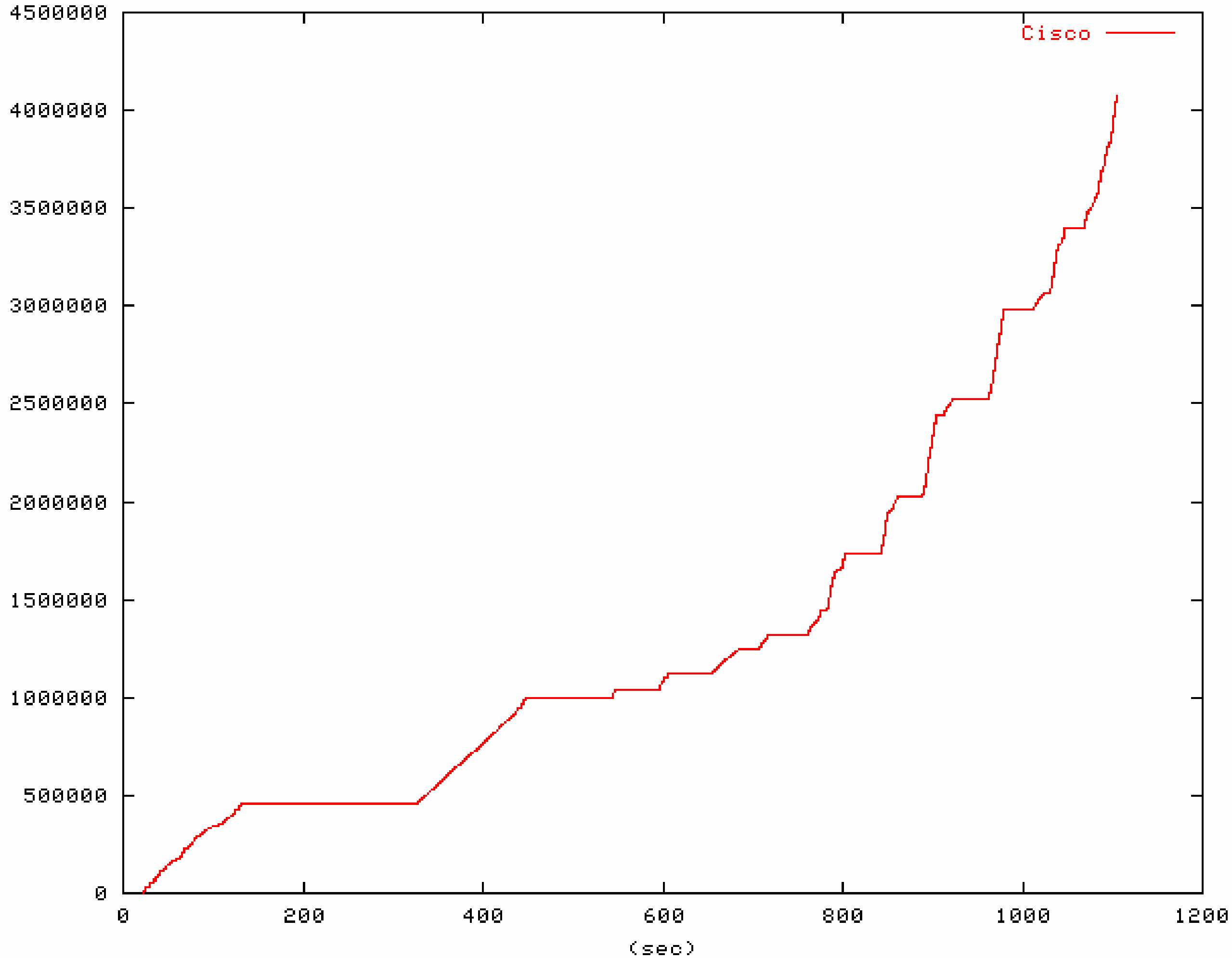
Measurement Result: 60 RRCs (zebra)



Measurement Result: 60 RRCs

- Convergence time comparison:
  - Cisco ⇒ 262 sec
  - Zebra⇒ never converge…
- Why zebra does not converge?
  - Shortage of main memory (512MB)
    - Limitation of PC based router performance
  - Cisco can converge even if 256MB memory
  - ⇨ Efficient memory management

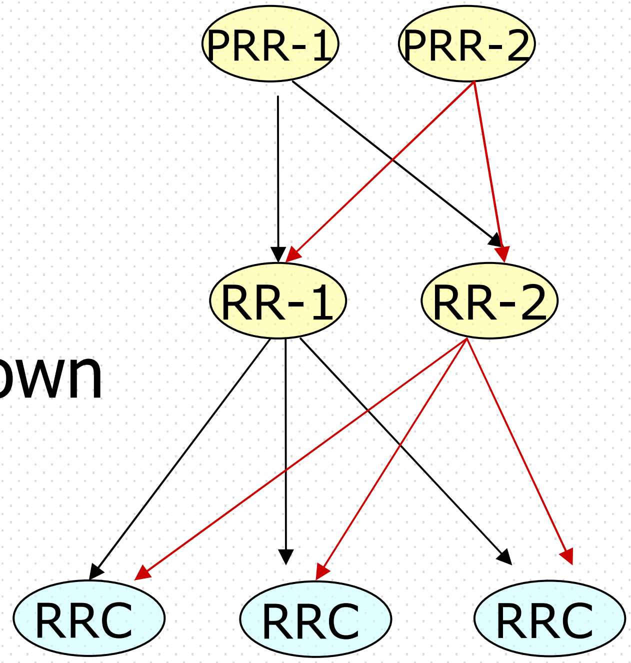
Measurement Result: 170 RRCs (Cisco)



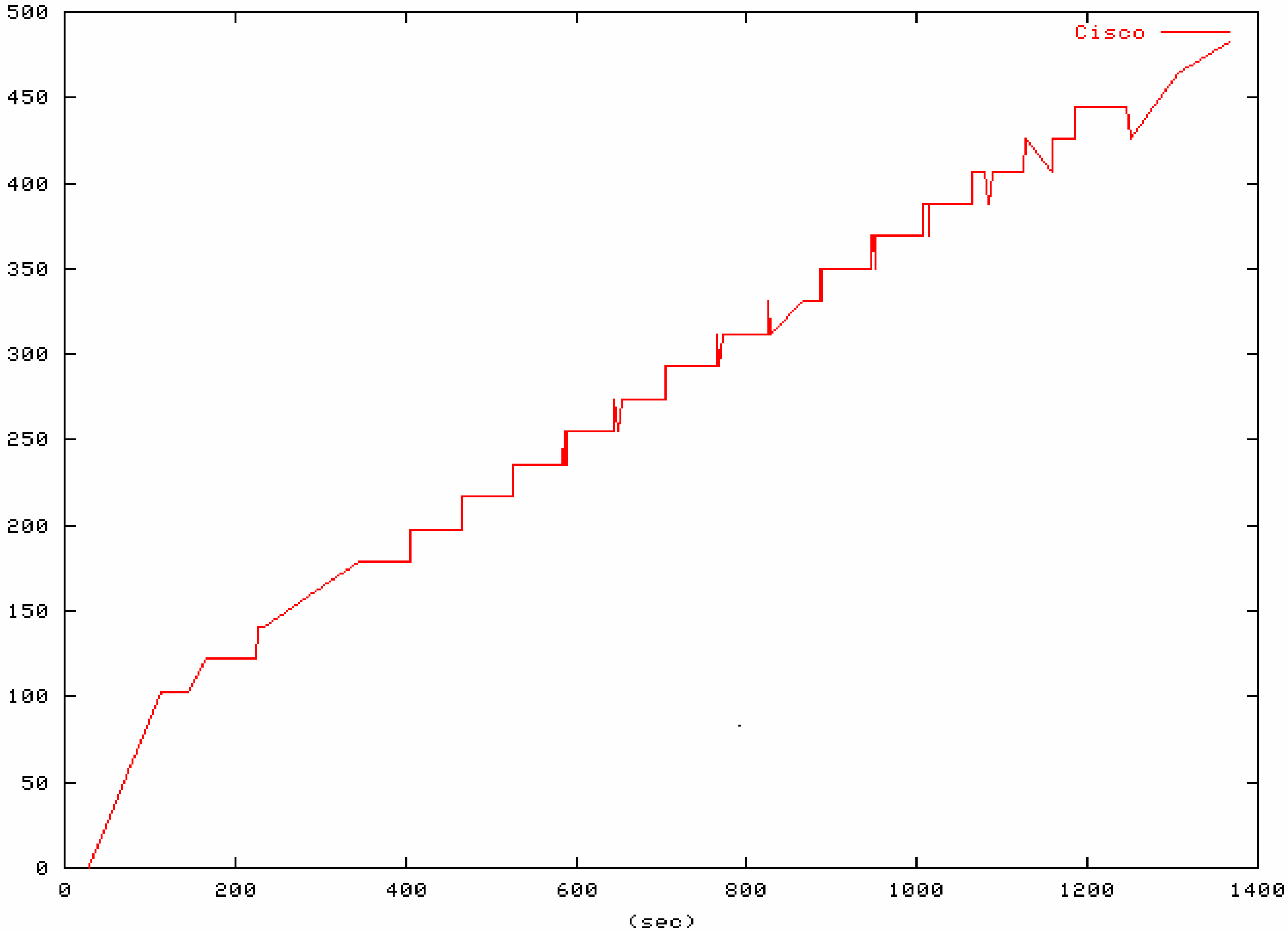
Measurement Result: 170 RRCs (Cisco)

- Convergence: 1150sec
- What if
  - Both PRR-1,PRR-2 are down

  At the same time
  - Then, restart



Measurement Result: 170 RRCs (Cisco)



Measurement Result: 170 RRCs (Cisco)

- Never converged:

Neighbor	V	AS	MsgRcvd	MsgSent	InQ	OutQ
172.16.0.62	4	65535	9	44744	0	291
172.16.0.63	4	65535	9	46217	0	319
172.16.0.64	4	65535	9	46310	0	724
172.16.0.65	4	65535	9	37370	0	169
172.16.0.66	4	65535	9	46374	0	665
172.16.0.67	4	65535	9	23387	0	125
172.16.0.68	4	65535	9	19541	0	0
172.16.0.69	4	65535	9	32036	0	0
172.16.0.70	4	65535	9	22729	0	306

- Why?
  - high overload in RR-1
    - Receive from both PRR-1,2 and Send update to RRC x 170
    - Limitation of CPU processing
    - Missing BGP update packet processing
    - Never finalize sending BGP update
    - Stack output queue

Conclusion

- Is this redundant route reflector architecture truly scalable?
  - When physical threshold turns over, it is never converged
    - Hierarchal Redundant RR architecture provide poor scalability
  - PC based router (zebra)
    - Performance depends upon main memory
  - Commercial router (Cisco)
    - Limitation of CPU processing

Future Research Direction

1. Better Route Reflector Architecture
   - Cascade update v.s. Route Reflector
2. Further BGP related measurement
   - More complicated topology
   - Other BGP technique e.g. route flap dampening