Routing and traffic measurements in ISP networks

Steve Uhlig

Network Architectures and Services
Delft University of Technology

Email : S.P.W.G.Uhlig@ewi.tudelft.nl
URL : http://www.nas.ewi.tudelft.nl/people/Steve/

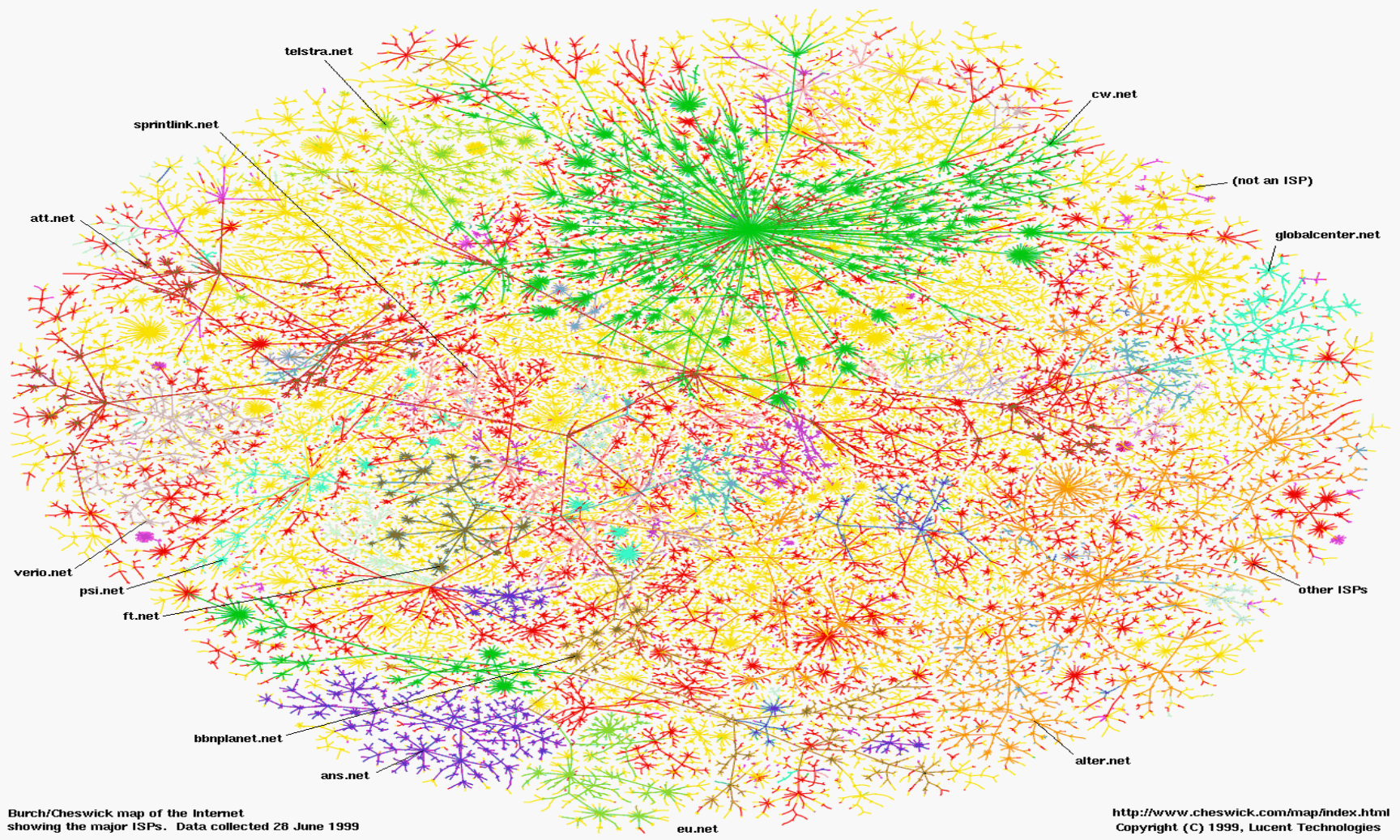
TU Delft

Delft University of Technology

Outline

- Organization of Internet routing 
- Organization of an AS
- Measuring the routing of an AS
- Measuring traffic in an AS
- Combining routing and traffic measurements
- Conclusions

A map of the Internet in 2000



Delft University of Technology

Organization of Internet Routing

- More than 21,000 autonomous routing domains:
  A domain is a set of routers, links, hosts and local area networks under the same administrative control. Domains are also called "autonomous systems" (AS).
- Domains size: from 1 PC to millions of hosts
- Domains are interconnected in various ways

Types of domains

- Transit domains:
  A transit domain allows external domains to use its own infrastructure to send packets to other domains
- Implicit hierarchy of transit domains according to "size"
- Examples: AT&T, UUNet, Level3, Opentransit, KPN,...

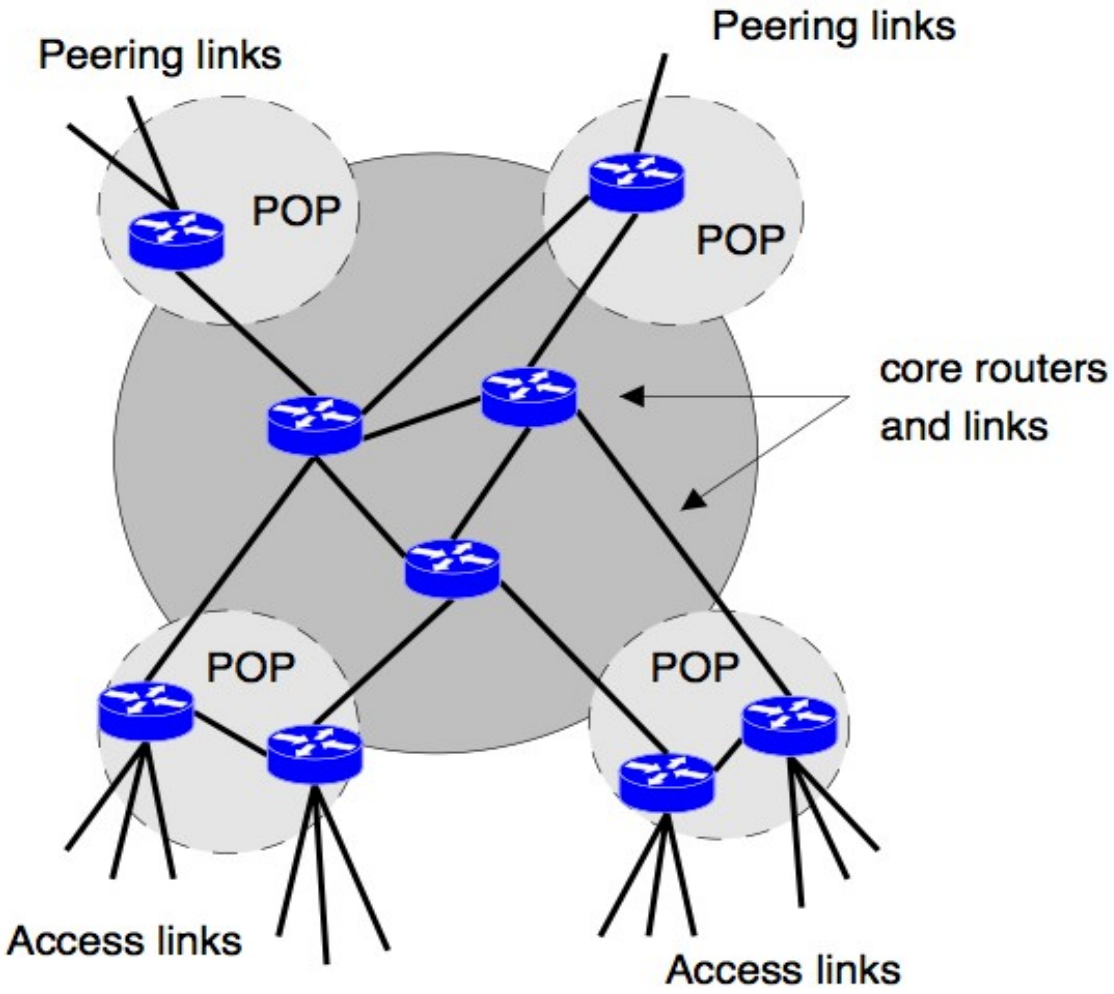
Types of domains

- Stub domains:
  A stub domain does not allow external domains to use its infrastructure to send packets to other domains
- A stub is connected to at least one transit domain
- Content stub domains: Yahoo, Google, MSN, BBC,...
- Access stub domains: ISPs providing Internet access via CATV, DSL,...

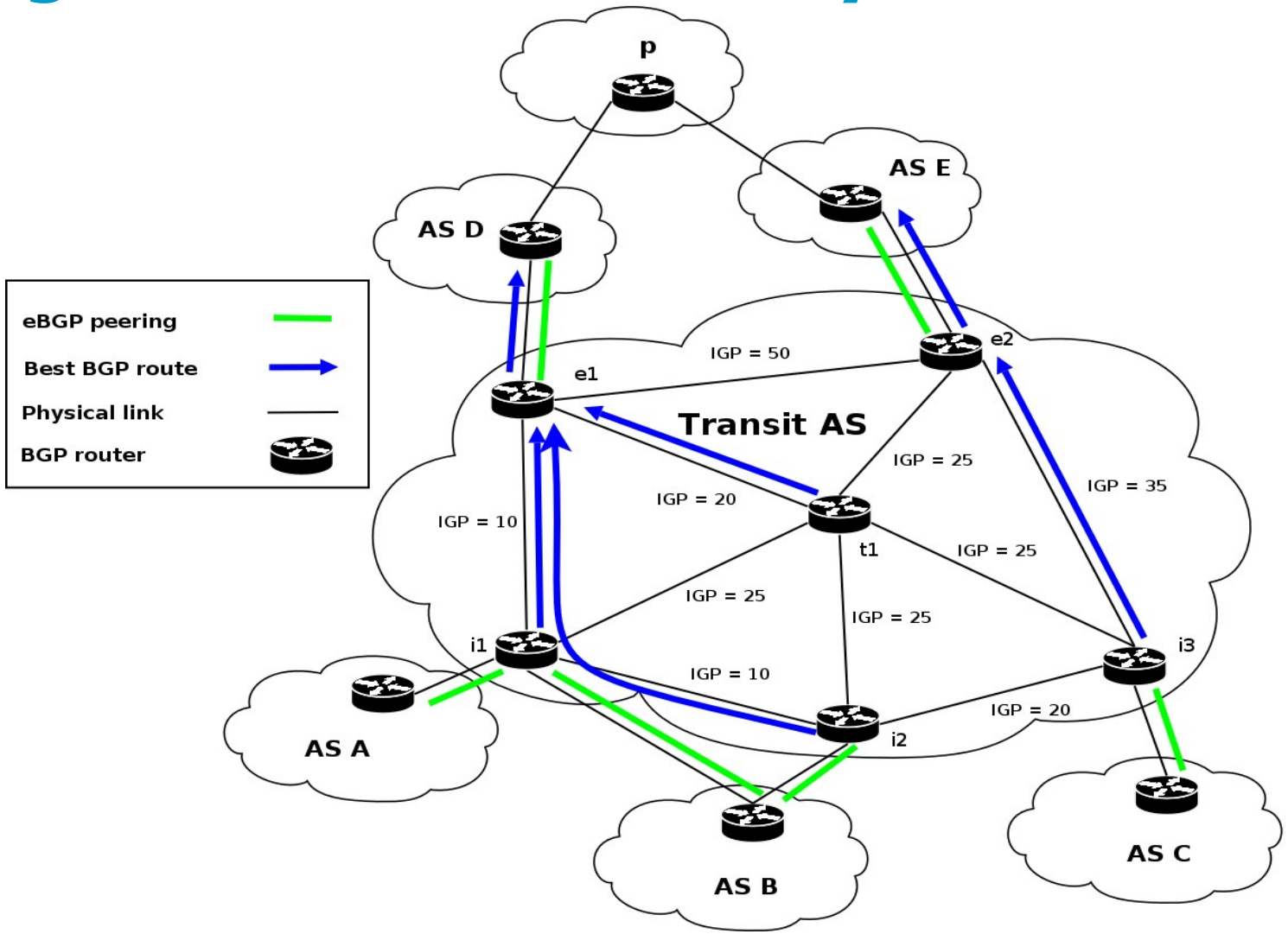
Outline

- Organization of Internet routing
- Organization of an AS 
- Measuring the routing of an AS
- Measuring traffic in an AS
- Combining routing and traffic measurements
- Conclusions

Structure of an Autonomous System



Routing in an Autonomous System



Intra- and inter-domain routing

- Interior Gateway Protocol (IGP):
  - Routing of IP packets inside each domain
  - Only knows topology of its domain

- Exterior Gateway Protocol (EGP):
  - Routing of IP packets between domains
  - Each domain is considered as an atomic structure

Intra-domain routing

- Goal: allow routers to transmit IP packets along the best path towards their destination
  - best usually means the shortest path
  - Allow to find alternate routes in case of failures
- Behavior: all routers exchange routing information
  - Each domain router can obtain routing information for the whole domain
  - The network operator or the routing protocol selects the cost of each link

Types of IGPs

- Static routing: only useful in very small domains
- Distance vector routing:
  - Routing Information Protocol (RIP)
    - Still widely used in small domains despite its limitations
- Link-state routing:
  - Open Shortest Path First (OSPF): widely used in enterprise networks
  - Intermediate System- Intermediate-System (IS-IS): widely used by ISPs

Inter-domain routing

- Goal: allow to transmit IP packets along the best path towards their destination
  - From an interdomain viewpoint, best path often means *cheapest* path
- Behavior:
  - Each domain specifies inside its routing policy the domains for which it agrees to provide a transit service and the method it uses to select the best path to reach each destination
  - Each router of the domain chooses its best path according to the routing policies, and advertises them to its neightboring routers

Inter-domain routes redistribution

- Inside a domain (iBGP):
  - Goal: propagate the routes learned from neighbors to the routers inside the domain
  - Implementation: full-mesh between BGP routers, route-reflection, or confederations
- Between domains (eBGP):
  - Goal: propagate external reachability to neighbors
  - Implementation: private peerings, public interconnection points

Organization of iBGP sessions

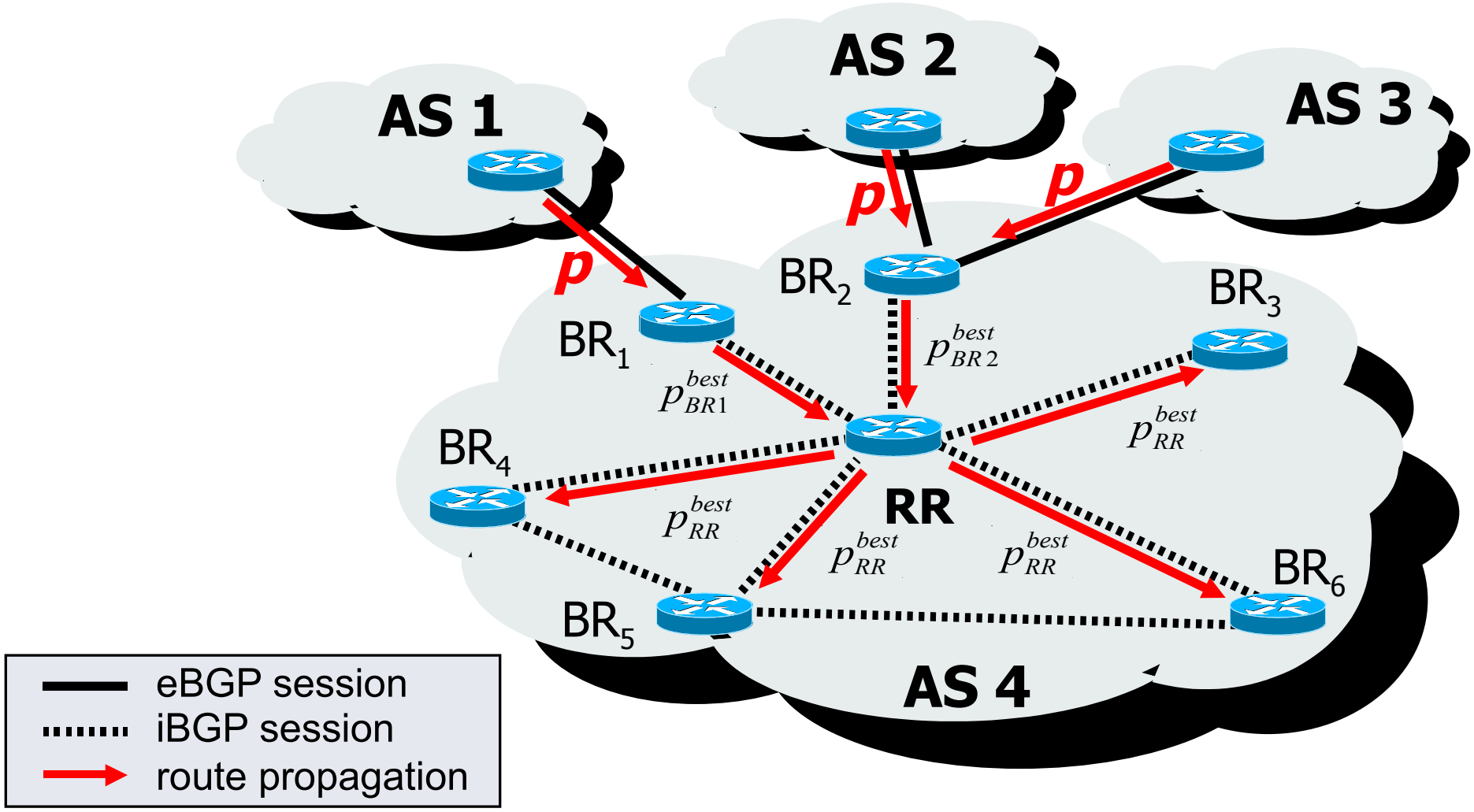
- iBGP full-mesh:
  - Pro's: full visibility of external routes, small convergence time
  - Con: N*(N-1)/2 iBGP sessions
- Route-reflection:
  - Pro: # iBGP sessions ~ # physical links
  - Con's: opaqueness of best route selection, slow convergence, possible route oscillations

For more details:

A. Basu, C. Ong, A. Rasala, B. Shepherd, and G. Wilfong. *Route oscillations in iBGP with route Reflection.* ACM SIGCOMM 2002.

T. Griffin and G. Wilfong. *On the correctness of iBGP configuration.* ACM SIGCOMM 2002.

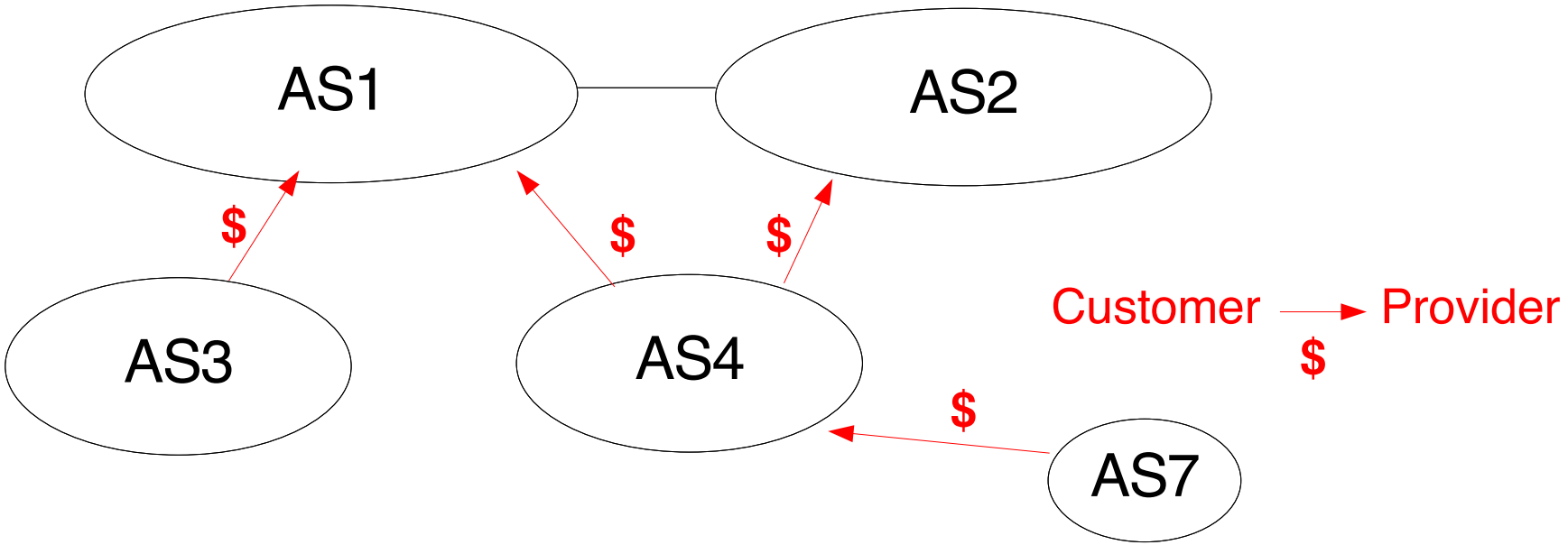
iBGP routes propagation



Routing policies

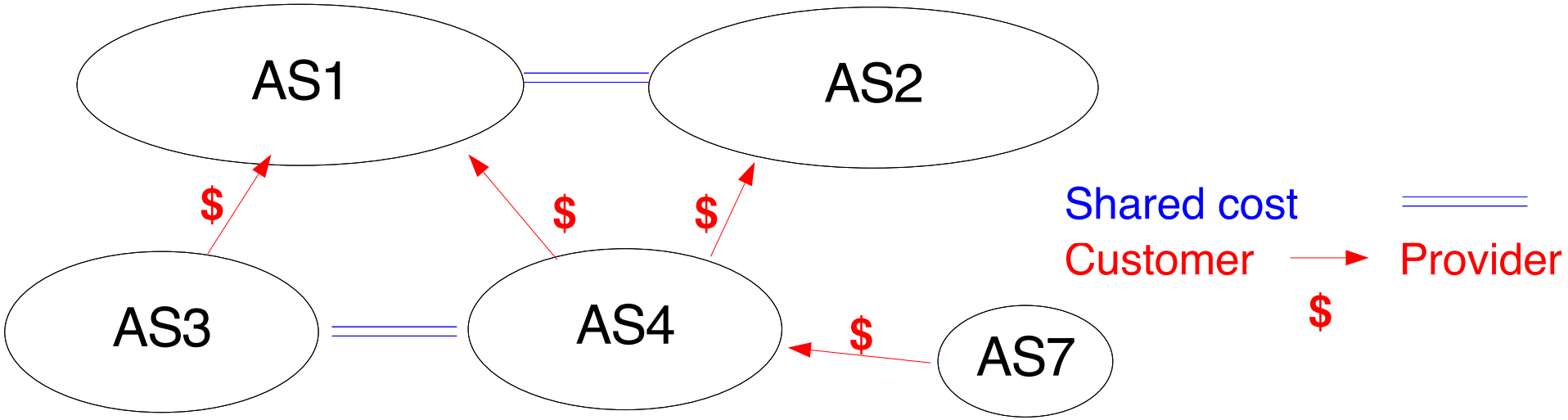
- In theory, BGP allows each domain to define its own routing policy
- In practice, there are two common policies:
  - customer-provider peering
    - Customer C buys Internet connectivity from provider P
  - shared-cost peering
    - Domains x and y agree to exchange packets by using a direct link or through an interconnection point

Customer-provider peering



- Customer sends to its provider its internal routes and the routes learned from its own customers => Provider will advertise those routes to the entire Internet to allow anyone to reach the Customer
- Provider sends to its customers all known routes => Customer will be able to reach anyone on the Internet

Shared-cost peering

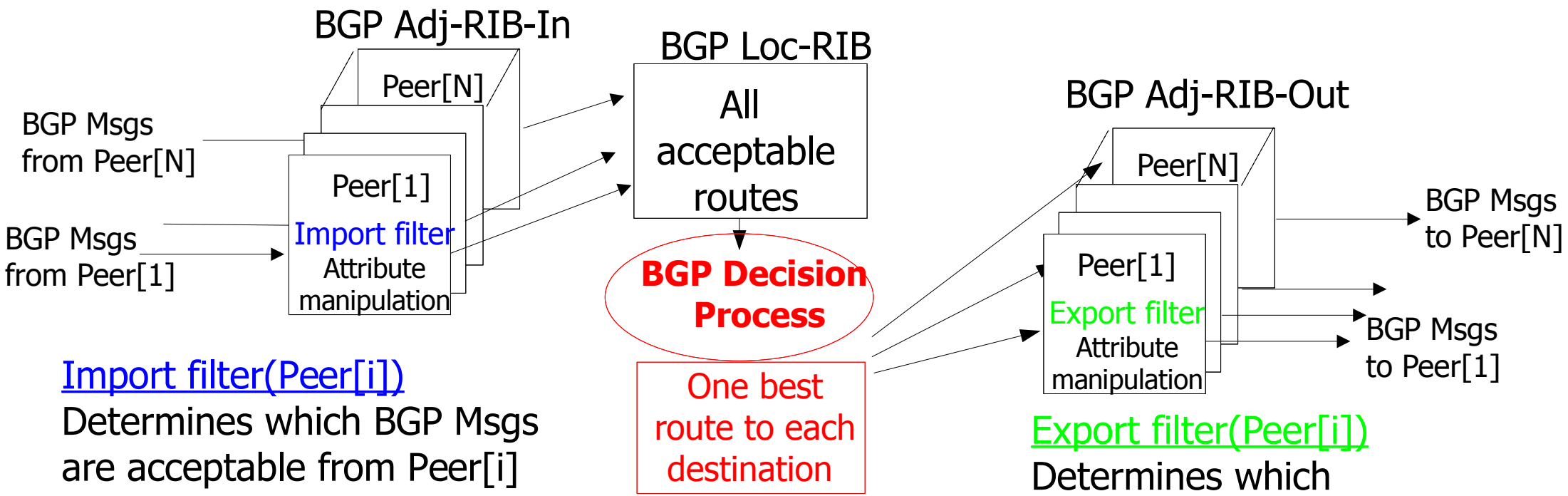


- Peer X sends to Peer Y its internal routes and the routes learned from its own customers
  - Peer Y will use shared link to reach Peer X and Peer X's customers
  - Peer X's providers are not reachable via the shared link
- Peer Y sends to Peer X its internal routes and the routes learned from its own customers
  - Peer X will use shared link to reach Peer Y and Peer Y's customers
  - Peer Y's providers are not reachable via the shared link

Routing policies

- Routing policies implement business relationships between domains
- The routing policy of a domain is implemented via the route filtering mechanism on BGP routers:
  - Inbound filtering: Upon reception of a route from a peer, a BGP router decides whether the route is acceptable, and if so whether to change some of its attributes.
  - Outbound filtering: Before sending its best route towards a destination, a BGP router decides which peers should receive this route and whether to change some of its attributes before sending it.

Conceptual operation of a BGP router



Import filter(Peer[i])
Determines which BGP Msgs are acceptable from Peer[i]

Export filter(Peer[i])
Determines which routes can be sent to Peer[i]

BGP Routing Information Base
Contains all the acceptable routes learned from all Peers + internal routes

- BGP decision process selects *the* best route towards each destination

Outline

- Organization of Internet routing
- Organization of an AS
- Measuring the routing of an AS 
- Measuring traffic in an AS
- Combining routing and traffic measurements
- Conclusions

Measuring the routing of an Autonomous System

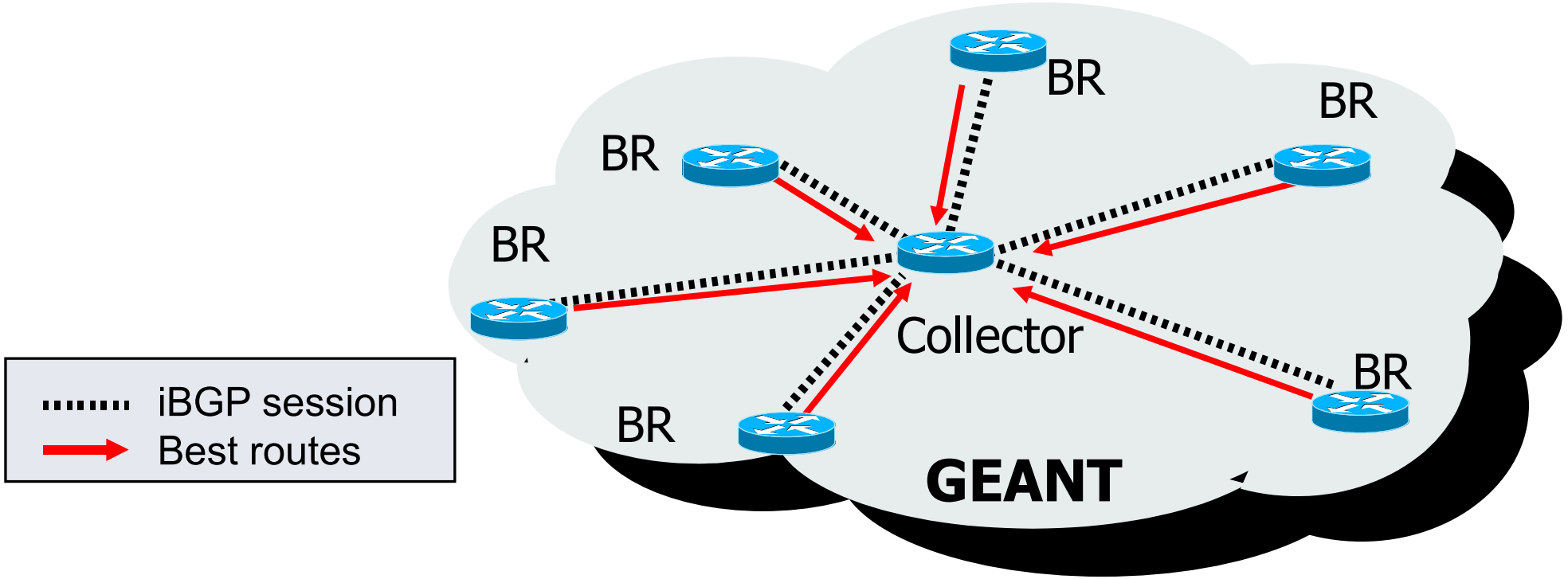
- IGP:
  - Flooding implies full topological visibility: one probe inside the whole domain is enough
  - 2 possibilities: capture LSPs flooded on an interface or through an IGP router
- BGP:
  - Lack of topological visibility due to path-vector protocol requires several capture points
  - 2 possibilities: capture external routes received from neighbors or iBGP routes

Measuring BGP routing

- eBGP capture:
  - Requires dumping all eBGP messages received from neighbors on border routers
  - High burden on CPU routers
  - Full visibility of BGP routes known
  - Impractical besides for BGP session debugging
- iBGP capture:
  - Configure one or several BGP routers to dump their RIB-ins or best routes
  - Misses full diversity known by the border routers
  - Always used in practice

Measuring the routing of an AS: GEANT

- IGP: ISIS capture through passive LSP monitoring (e.g. pyrt)
- BGP: iBGP capture with one router inside the iBGP full-mesh



Measuring the routing of an AS: GEANT

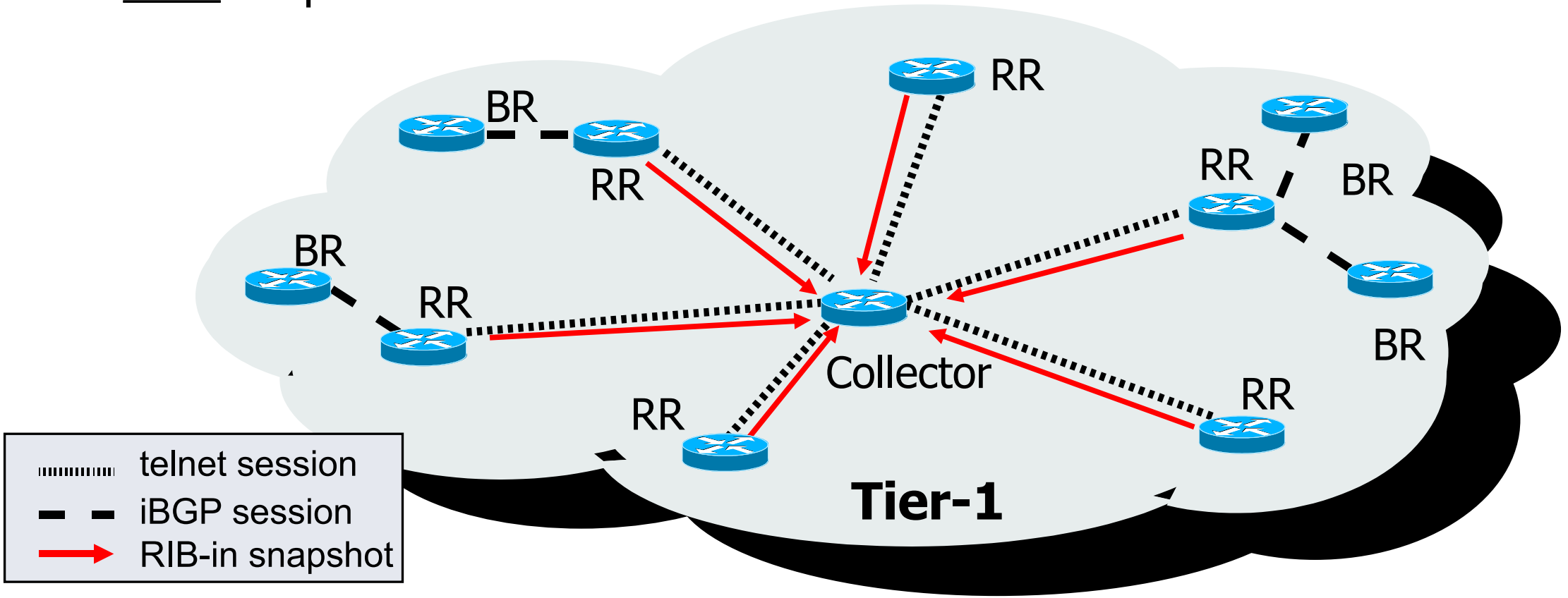
- IGP:
  - All topological changes inside IGP are fully visible
- BGP:
  - Part of the multiple routes known to border routers are unknown inside the iBGP mesh
  - All best routes are known
  - Dynamics of best routes is known

For more details:

B. Quoitin and S. Uhlig. Modeling the routing of an Autonomous System with CBGP. *IEEE Network magazine, November/December 2005.*

Measuring the routing of an AS: tier-1

- IGP: ISIS capture through passive LSP monitoring
- BGP: snapshots of RIB-ins from main route-reflectors



Measuring the routing of an AS: tier-1

- IGP:
  - Complete visibility of topological changes in the network
- BGP:
  - Multiple routes at border routers are lost
  - Partial visibility of best routes known to AS (RR redistribution rules)
  - No visibility of dynamics

For more details:

S. Uhlig and S. Tandel. *Measuring the route diversity inside a tier-1 network.* Proc. of IFIP Networking conference, Coimbra, Portugal, May 2006.

Outline

- Organization of Internet routing
- Organization of an AS
- Measuring the routing of an AS
- Measuring traffic in an AS 
- Combining routing and traffic measurements
- Conclusions

Measuring the traffic of an AS

3 main approaches:

- Link-level traffic measurements:
  - SNMP statistics: total bytes/packets every 5 minutes
- Flow-level statistics:
  - Netflow: flow caching and export
- Packet-level statistics:
  - Libpcap, DAG: headers or full packets

Link-level traffic measurements

Principle:

- Get every 5 minutes number of bytes and packets that passed on each link of the network
- To obtain the traffic matrix (OD pairs), invert the link-level measurements to obtain the traffic matrix: $y = Ax$ where A is the routing matrix, y are the link-level measurements, and x the traffic matrix.
- The link-level measurements (y) are actually the outcome of routing (A) applied to OD flows (x)

Link-level traffic measurements

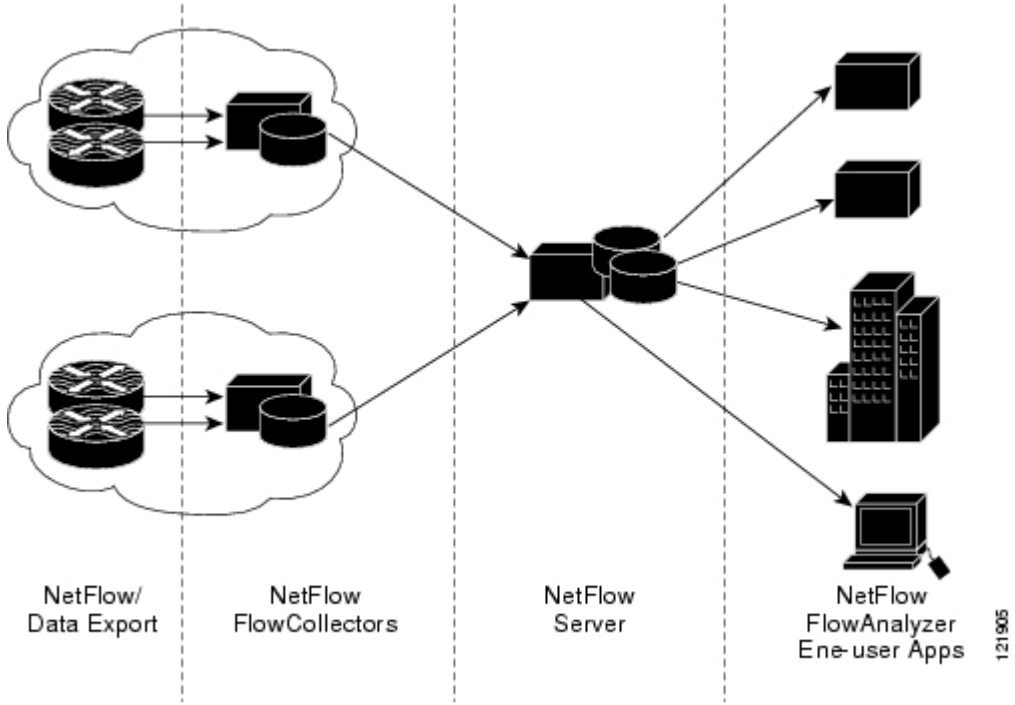
- Inverting $y = Ax$ is in practice an underconstrained linear inverse problem with an infinity of solutions
- $x \sim O(N^2)$ where N is the number of nodes while typically $y \sim O(N)$
- Side information has to be added, usually in the form of constraints on the distribution of the elements x (Poisson, gravity model) or temporal constraints on the time evolution (Kalman filtering)
- Still, getting SNMP data does not cost much

Flow-level traffic measurements

- Principle:
  - At each ingress interface in the network, monitor all incoming flows and report them to a collector
  - Cisco's Netflow was originally a caching system for IP packets forwarding
  - Netflow evolved as a system for network-wide flow monitoring and IP accounting
  - Netflow is becoming a de-facto standard for network-wide monitoring

Netflow: architecture

- The NetFlow cache or data source which stores IP Flow information
- The NetFlow export or transport mechanism that sends NetFlow data to a network management collector for data reporting



Netflow: flows

- A flow is identified as a *unidirectional* stream of packets between a given source and destination
- A flow is a 7-key tuple: <src IP, dst IP, src port, dst port, protocol type, ToS byte, input interface>
- These seven key fields define a unique flow. If a flow has one different field than another flow, then it is considered a new flow
- A flow contains other accounting fields (such as the AS number in the NetFlow export Version 5 flow format) that depend on the version record format that you configure for export

Netflow: cache and flow management

1. Create and update flows in NetFlow cache

SrcIf	Src Padd	DstIf	Dst Padd	Protocol	TOS	Flgs	11000	00A2	Src Msk	Src AS	00A2	Dst Msk	Dst AS	Next Hop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.023.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.023.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	00A1	/24	180	00A1	/24	15	10.023.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.023.2	1040	1745	14

2. Expiration

- Inactive timer expired (15 sec is default)
- Active timer expired (30 min (1800 sec) is default)
- NetFlow cache is full (oldest flows are expired)
- RST or FIN TCP flag

Srcif	Src Padd	Dstif	Dst Padd	Protocol	TOS	Flgs	11000	00A2	Src Msk	Src AS	00A2	Dst Msk	Dst AS	Next Hop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.023.2	1528	1800	4

3. Aggregation

No

Yes

4. Export version
   Non-Aggregated Flows—Export Version 5 or 9

5. Transport protocol

e.g. Protocol-Port Aggregation Scheme Becomes

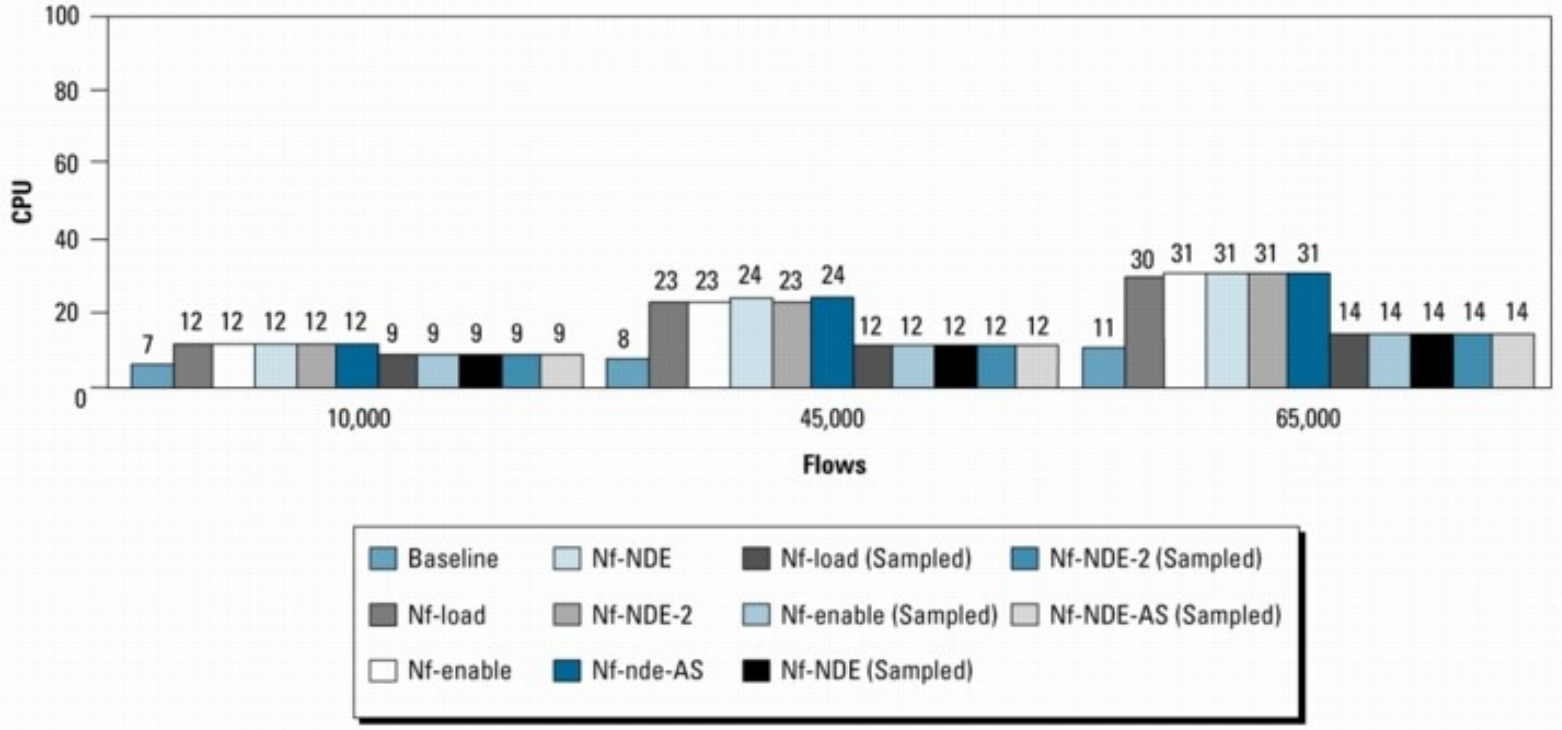
Protocol	Pkts	SrcPort	DstPort	Bytes/Pkt
11	11000	00A2	DstPort	1528

Aggregated Flows—Export Version 8 or 9

12910

Netflow: performance

- Unless amount of traffic is small, sampling has to be used to spare CPU, memory and storage space
- CPU usage on high-end router (Cisco 12000):

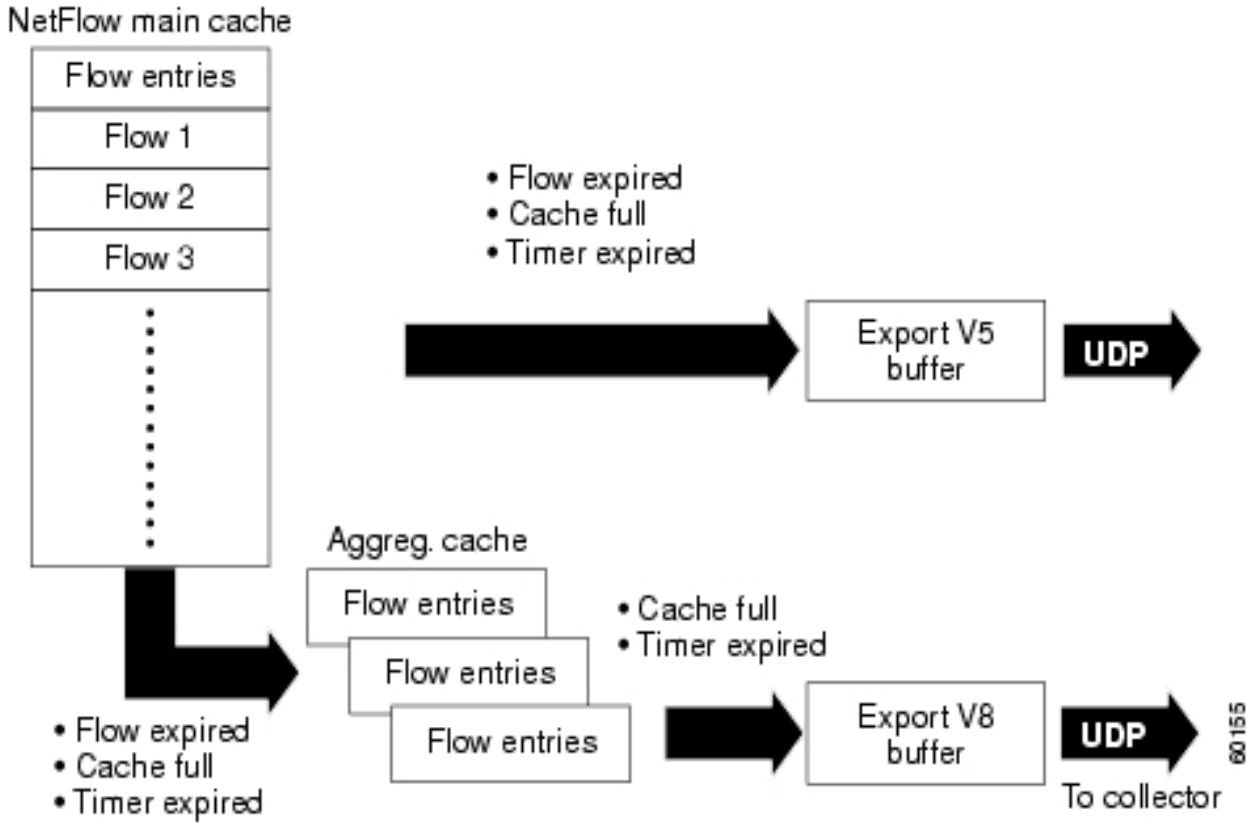


Netflow: performance

- Besides CPU, exporting Netflow statistics requires a lot of storage space
- Example: In GEANT, one month of gzipped Netflow data consumes ~ 150 Gbytes of disk space
- In a large ISP, activating Netflow and storing each flow would amount to several TBytes of data per month (unless aggregation is used)
- Processing and analyzing is very time-consuming
- Juniper does not provide customer support under 1/1000 Netflow sampling rate

Netflow: performance

- Limiting amount of storage space is possible through flow aggregation:



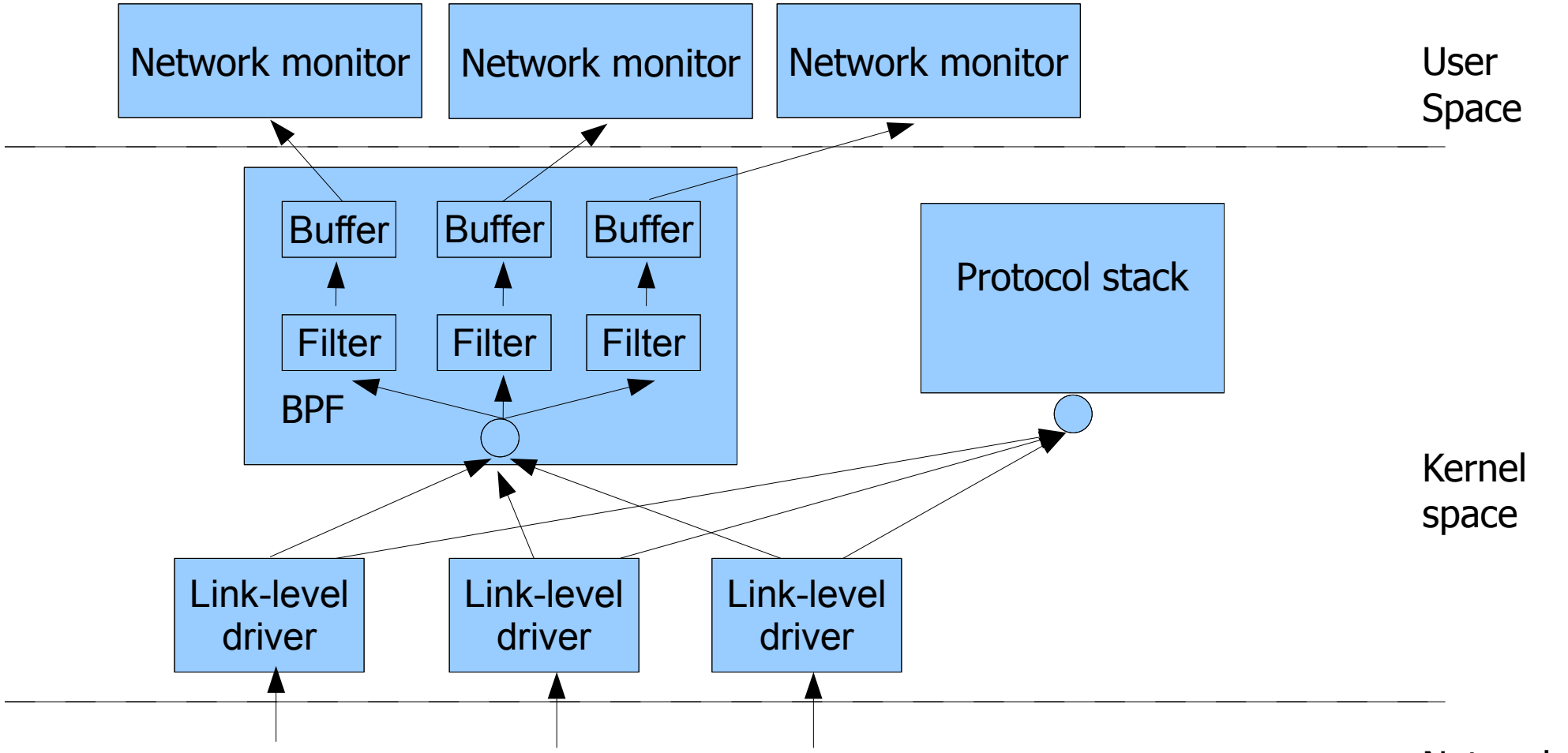
Packet-level traffic measurements

Principle:

- Put the NIC in promiscuous mode to see all packets (requires root privilege)
- Capture all raw packets seen by the NIC and copy them on disk
- Requires special library on the host to access the raw packets from the NIC (libpcap, DAG,...)

- Examples of packet sniffers: tcpdump, ethereal, snoop, Windows network monitor,...

Packet-level traffic measurements



Packet-level traffic measurements: performance

- Since network monitoring applications are working in user-space, copying packets to disk is typically slow
- Moving the network monitor applications to kernel space (DAG tools, nprobe) may limit the performance issues
- Still, packet-level statistics require a lot of storage space depending on what fraction of the packets is stored and do not scale to network-wide measurements

Outline

- Organization of Internet routing
- Organization of an AS
- Measuring the routing of an AS
- Measuring traffic in an AS
- Combining routing and traffic measurements
- Conclusions

Combining routing and traffic measurements

- Combining routing and traffic requires to build a model of the network
- Modeling a network requires to integrate 4 parts:
  - Router configurations
  - Topology
  - Routing
  - Traffic

Router configurations

- Mapping between physical topology and logical one (IP)
- IGP weights
- IGP structure (areas)
- iBGP structure (full-mesh, route-reflection, confederation)
- eBPG peerings and policies (filtering, attributes manipulation, BGP timers)

Topology

- What level of detail to use?
  - POP-level?
  - Router-level?
  - IP-interfaces-level?
- Granularity depends on what information you have (configs) and what you want to achieve...
- Loopback interfaces issues: IGP and BGP might use different IP's to refer to a next hop, while this might be the same router

Routing data

- IGP:
  - Any router sees all topological changes: a simple Dijkstra is a very good model of IGP paths
  - If OSPF areas are used, it's a bit sutble how to faithfully reproduce the path choices
- BGP:
  - Routers have a very limited view of the BGP routes known inside the AS
  - Getting RIB-in's from all routers is painful
  - Next-hop-self and external peerings are often a nightmare
  - Not much hope to have more than a rough approximation of the real routers' RIB-ins

Traffic data

- Aggregate flows according to network topology model (interface, router, POP)
- Transform flows into traffic demand, e.g. <ingress, destination, amount of traffic>
- Route traffic using simulation model to replay traffic on network topology
- Traffic statistics granularity has to match the network model

Putting it all together

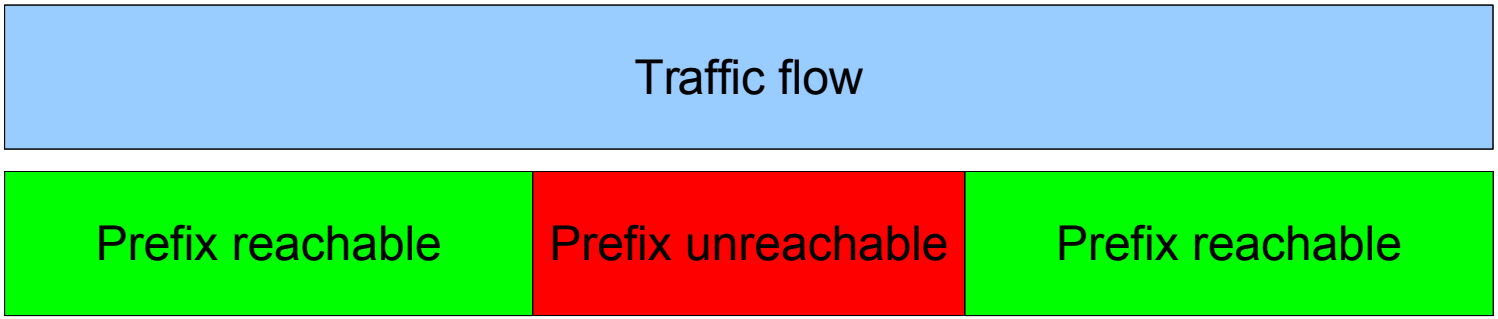
- Building an intradomain traffic matrix:
  - use SNMP data and infer it
  - rely on Netflow + routing + configs to get per-prefix traffic matrix

    For more details:

    S. Uhlig, B. Quoitin, S. Balon and J. Lepropre. *Providing public intradomain traffic matrices to the research community.* ACM SIGCOMM Computer Communication Review, 36(1):83-86, January 2006.
- Building an interdomain traffic matrix:
  - per-prefix traffic statistics necessary (Netflow)
  - Size of traffic statistics and handling routing model are key issues

The time issue

- Traffic statistics span time intervals during which traffic was observed (considering flow timeouts of traffic capture)
- Routing data sees changes in topological/reachability information
- How should one attribute traffic to time bins?



Simulating routing and traffic

- Reproducing routing dynamics is not scalable with discrete-event simulators (ns-2, SSFNet, JSim)
- Ad-hoc simulators are required for scalability (e.g. CBGP)
- How to correctly attribute traffic to a routing path given that time granularity (routing/traffic) and visibility (routing) is unclear ?
- Do we care at all about what happens between visible changes?
- So far, people have assumed that we should not care

Outline

- Organization of Internet routing
- Organization of an AS
- Measuring the routing of an AS
- Measuring traffic in an AS
- Combining routing and traffic measurements
- Conclusions

Conclusions

- Understanding routing and traffic measurements requires properly understanding the measured system
- Visibility of routing is a clear issue for interpretation of traffic measurements
- Trade-off:
  - Finer granularity of measurements hopefully brings more accurate picture
  - Cost/benefit of improved measurements granularity unclear