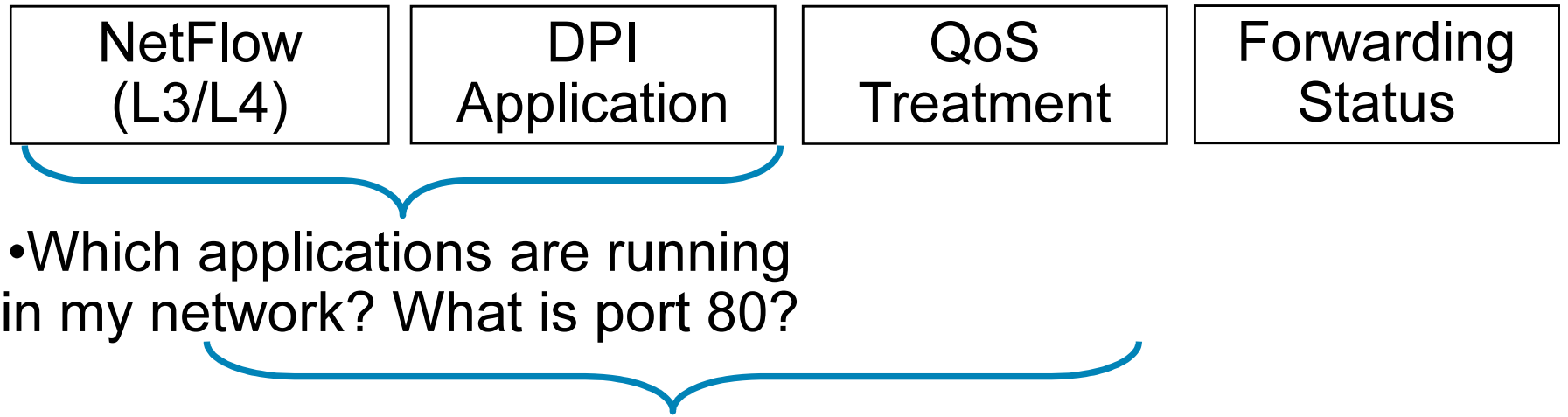
NetFlow/IPFIX Various Thoughts

Paul Aitken & Benoit Claise

3rd NMRG Workshop on NetFlow/IPFIX Usage in Network Management, July 2010

#1 Application Visibility Business Case



- Next requirement: performance metric (link with PMOL)

Flexible Flow Record: Key Fields

Flow
Sampler ID
Direction
Interface
Input
Output
Layer 2
Source VLAN
Destination VLAN
Source MAC address
Destination MAC address

IPv4	
IP (Source or Destination)	Payload Size
Prefix (Source or Destination)	Packet Section (Header)
Mask (Source or Destination)	Packet Section (Payload)
Minimum-Mask (Source or Destination)	TTL
Protocol	Options bitmap
Fragmentation Flags	Version
Fragmentation Offset	Precedence
Identification	DSCP
Header Length	TOS
Total Length	

IPv6	
IP (Source or Destination)	Payload Size
Prefix (Source or Destination)	Packet Section (Header)
Mask (Source or Destination)	Packet Section (Payload)
Minimum-Mask (Source or Destination)	DSCP
Protocol	Extension Headers
Traffic Class	Hop-Limit
Flow Label	Length
Option Header	Next-header
Header Length	Version
Payload Length	

Flexible Flow Record: Key Fields

Routing
src or dest AS
Peer AS
Traffic Index
Forwarding Status
IGP Next Hop
BGP Next Hop
Input VRF Name

Transport	
Destination Port	TCP Flag: ACK
Source Port	TCP Flag: CWR
ICMP Code	TCP Flag: ECE
ICMP Type	TCP Flag: FIN
IGMP Type*	TCP Flag: PSH
TCP ACK Number	TCP Flag: RST
TCP Header Length	TCP Flag: SYN
TCP Sequence Number	TCP Flag: URG
TCP Window-Size	UDP Message Length
TCP Source Port	UDP Source Port
TCP Destination Port	UDP Destination Port
TCP Urgent Pointer	

Application
Application ID*

Multicast
Replication Factor*
RPF Check Drop*
Is-Multicast

***: IPv4 Flow only**

- Near future: Class-map

Flexible Flow Record: Non-Key Fields

Counters
Bytes
Bytes Long
Bytes Square Sum
Bytes Square Sum Long
Packets
Packets Long

Timestamp
sysUpTime First Packet
sysUpTime First Packet

IPv4
Total Length Minimum (*)
Total Length Maximum (*)
TTL Minimum
TTL Maximum

IPv4 and IPv6
Total Length Minimum (**)
Total Length Maximum (**)

- Plus any of the potential “key” fields: will be the value from the first packet in the flow

(*) IPV4_TOTAL_LEN_MIN, IPV4_TOTAL_LEN_MAX
(**)IP_LENGTH_TOTAL_MIN, IP_LENGTH_TOTAL_MAX

#2 Mediation Function in Router

- Mediation: Data aggregation, reduction, correlation, and analysis

  Aggregation in space (different line cards in the router)

  Aggregation in time (performance metrics)

IPFIX export in branch offices
+ WAN export bandwidth limitation
+ performance metrics sent on regular basis for performance assurance
+ NetFlow export from different observation domains in the router
= mediation function + IPFIX structured data

#3 NetFlow as Alternative to syslog?

- Logging in high-performance environments is nontrivial, NetFlow is replacing syslog
- Flow event information can now be exported through NetFlow v9
  - Information about NAT modifications to the traffic
  - Information about Flows denied by security policy
  - Information about AAA/usernames associated with flows
- Provides scalable logging
  - 10-Gbps flows, 100-k connections per second = lots of logs
- Firewall: gain in terms of connection/s and throughout
  - Surprised by the gain
- NetFlow export is the logical evolution in logging technology?

#4 Performance Challenge Moving Bottleneck

B

- “consume a lot of CPU”
  -> packet sampling
  -> metering process in hardware
- “collision in the cache”
  -> improved the hash function
  -> increased the cache size
- “consume much bandwidth”
  -> flexible flow record
  -> per interface, per direction
  -> export cache type per collector
  -> flow selection method

- Next one: the collector? “Scalable and Robust Decentralized IP Traffic Flow Collection and Analysis (SCRIPT)” with the Zurich university

#5 Metering Process Challenge Flexible NetFlow is very Flexible…

P

- Easier to shoot yourself in the foot
- Let's not forget that the router still has to route packets
- Might need some consulting services for every customer

  No one size fits all

- Example

```
match datalink mac {destination address input |
    source address {input | output}}
```

#5 Metering Process Challenge Flexible NetFlow is very Flexible…

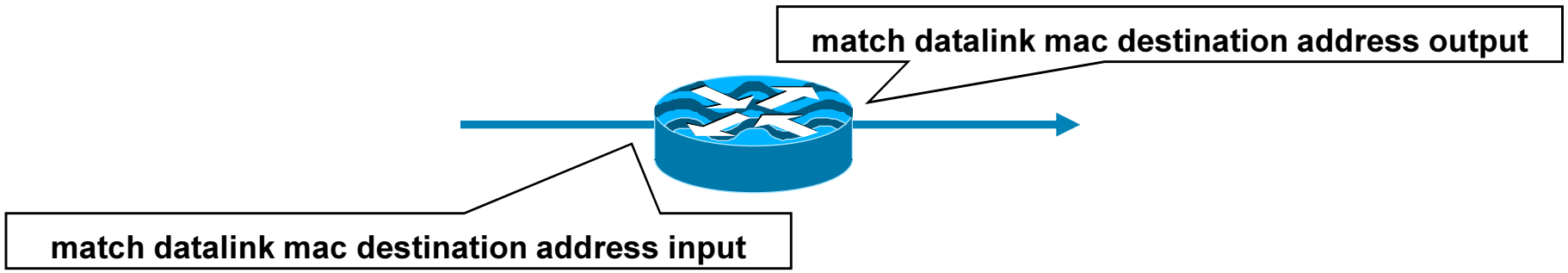
```
match datalink mac {destination address input | source
address {input | output}}
```

destination address input destination MAC address of packets received by the router

source address source MAC address

input Packets received by the router

output Packets transmitted by the router



Example of MPLS PE with QoS … and a distributed system

#6 Options Template Overload…

- Options Template is used for …

  Reducing Redundancy

  Statistics Information in the IPFIX protocol (The Metering Process Statistics, The Metering Process Reliability Statistics, Exporting Process Reliability Statistics)

  The Flow Keys Option Template

  ifIndex/interface name matching

- Yes everything is possible with Options Template Record
- This complicates the collecting process
- IPFIX Structured Data helps

  Example: reducing redundancy

#7 IPFIX Future Work?

- IPFIX Doctors
- IPFIX Applicability Version 2
- IPFIX Mediator protocol will have to be done
- Consistent application information export?

#8 Permanent Cache Type = user defined PUSH MIB B

- Next to normal and immediate cache types …
- Permanent cache
  - To track a set of flows without expiring the flows from the cache
  - Entire cache is periodically exported (update timer)
  - After the cache is full (size configurable), new flows will not be monitored
  - Uses update counters rather than delta counters
-

NetFlow/IPFIX	SNMP
Push	Pull
Regular export	Polling time
Information Element	SNMP OID
User Defined Information Element	No equivalent

NetFlow/IPFIX Various Thoughts

Paul Aitken & Benoit Claise

3rd EMANICS Workshop on Netflow/IPFIX Usage, July 2010