# Software Tool for Passive Real-Time Measurement of QoS Parameters

**Mihai Vlad, Ionut Sandu**, *Alcatel Timisoara, Romania*

**Virgil Dobrota, Ionut Trestian**, *T.U. of Cluj-Napoca, Romania*

**Jordi Domingo-Pascual**, *U.P.C. Barcelona, Spain*

# Agenda

1. **Introduction**
2. **QoS Parameters**
   OWD, AOWD, IPDV, AIPDV
3. **Designing of the Measurement Tool**
   Packet capturing
   Time-stamping
   Flow ID generation
   Classification
   Packet ID generation
   Measurement result transfer
4. **Proposed Architecture**
5. **Experimental Results**
6. **Conclusions**

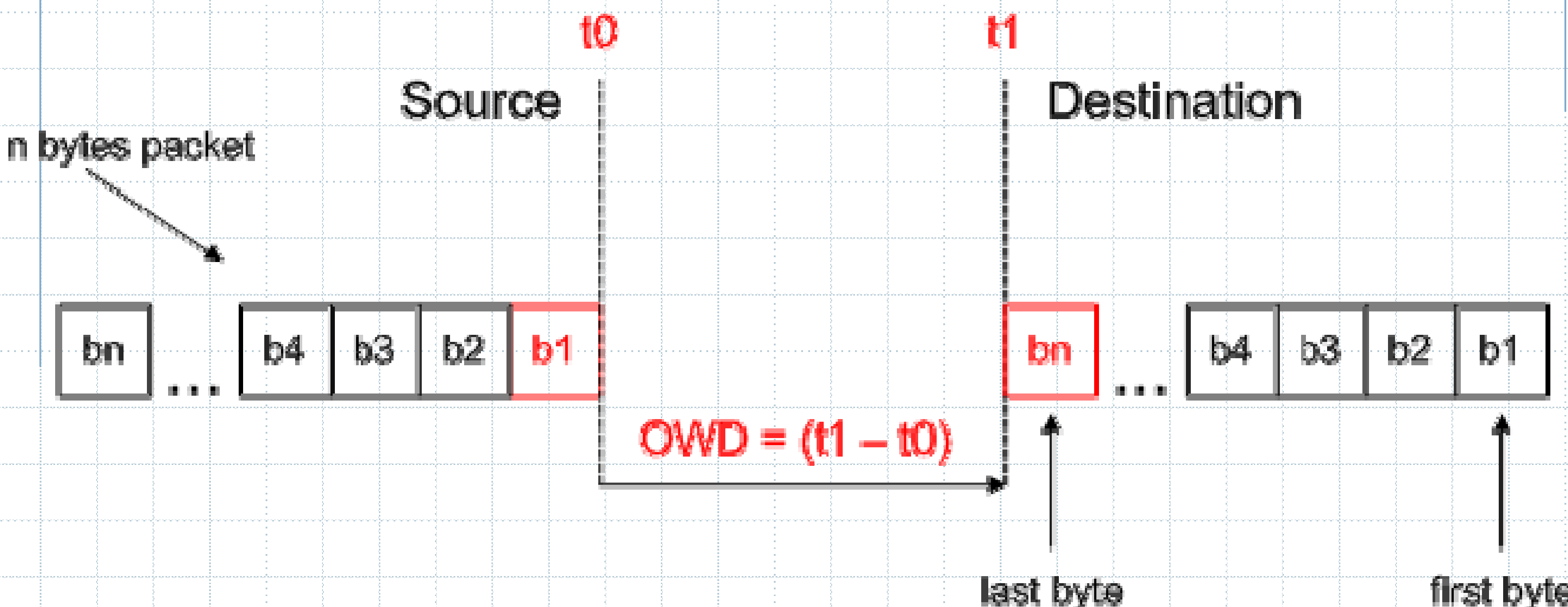
# 1. Introduction (I)

- New QoS standards for Heterogeneous Networks: **EuQoS** – FP6 European Project
- Developed at Universitat Politecnica de Catalunya, Barcelona (Socrates-Erasmus grant in 2005)
- Based on **OreNETa** (*One-way delay REaltime NETwork Analyzer*) – Abel Navaro
- **Real–time** network analysis
  - QoS parameters computed on-the-fly as average values over a period of time (e.g. one second)

# 1. Introduction (II)

- **Passive capturing advantages**
  - No additional traffic on the network
  - Packets are not modified
- **Heterogeneous Networks:** different transmission technologies between two flow endpoints
  - Ethernet/IEEE 802.3
  - Fast Ethernet/IEEE 802.3u
  - Wireless LAN (802.11b/g)
  - ATM
  - Gigabit Ethernet

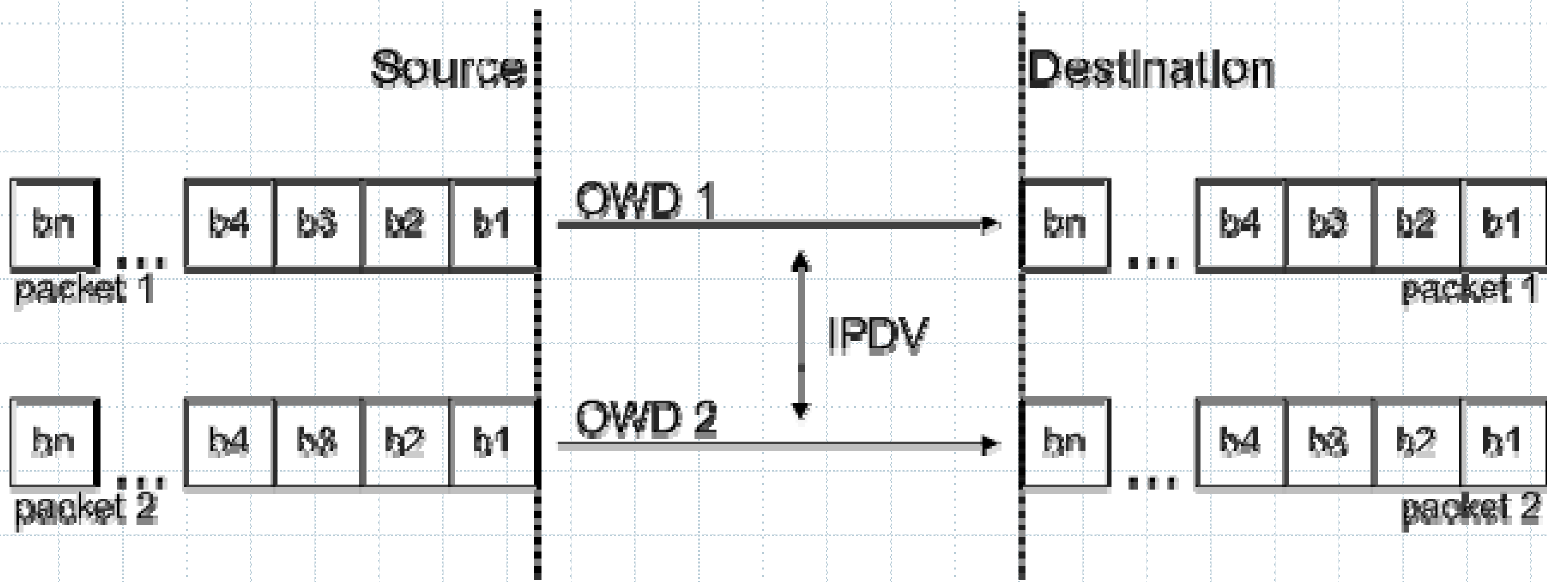
# 2. QoS Parameters: (Average) One Way Delay




$$OWD_i = t_{1i} - t_{0i} \quad 1 < i \leq N$$

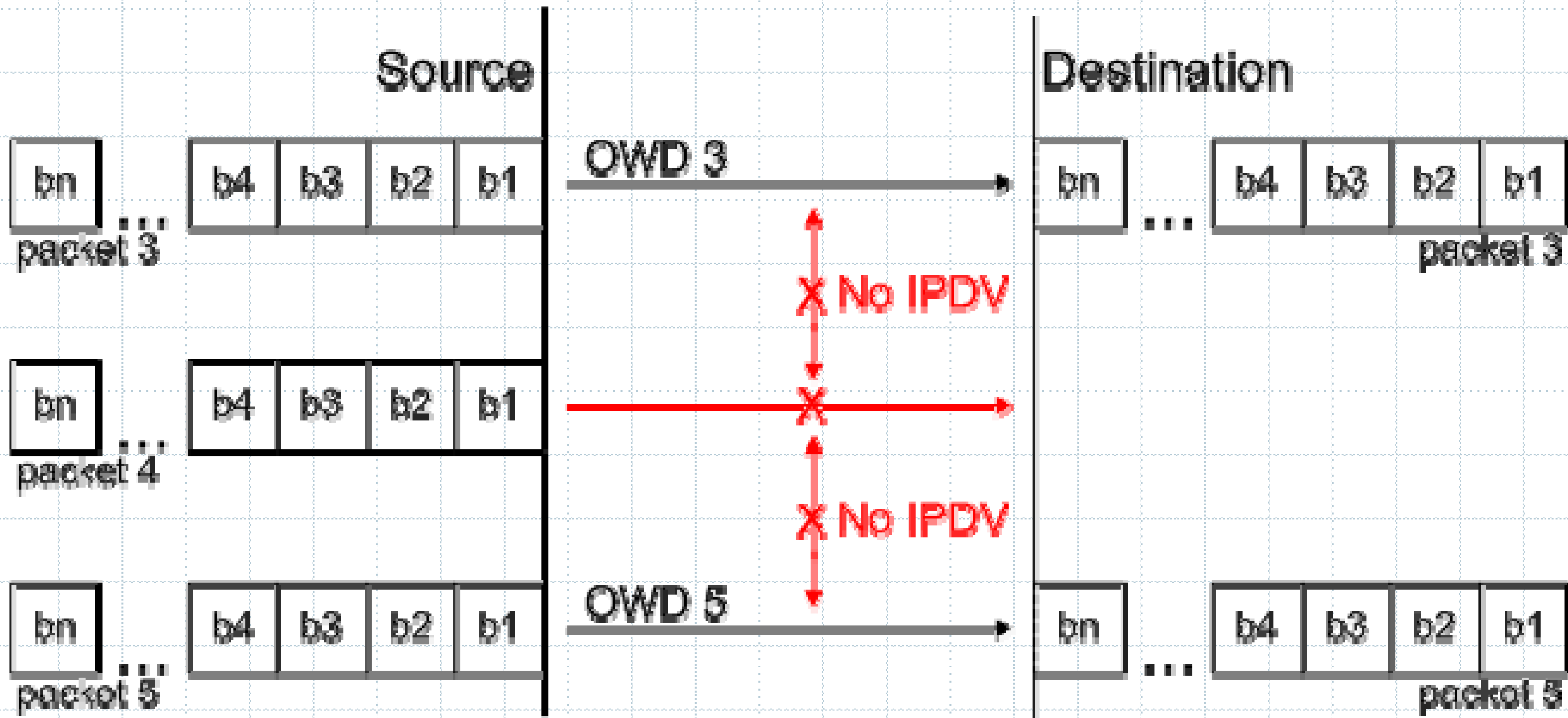
$$AOWD = \frac{\sum_{i=1}^{N} OWD_i}{N}$$

# 2. QoS Parameters: IP Packet Delay Variation (I)




$$IPDVi = OWD(i-1) - OWDi$$

# 2. QoS Parameters: IP Packet Delay Variation (II)



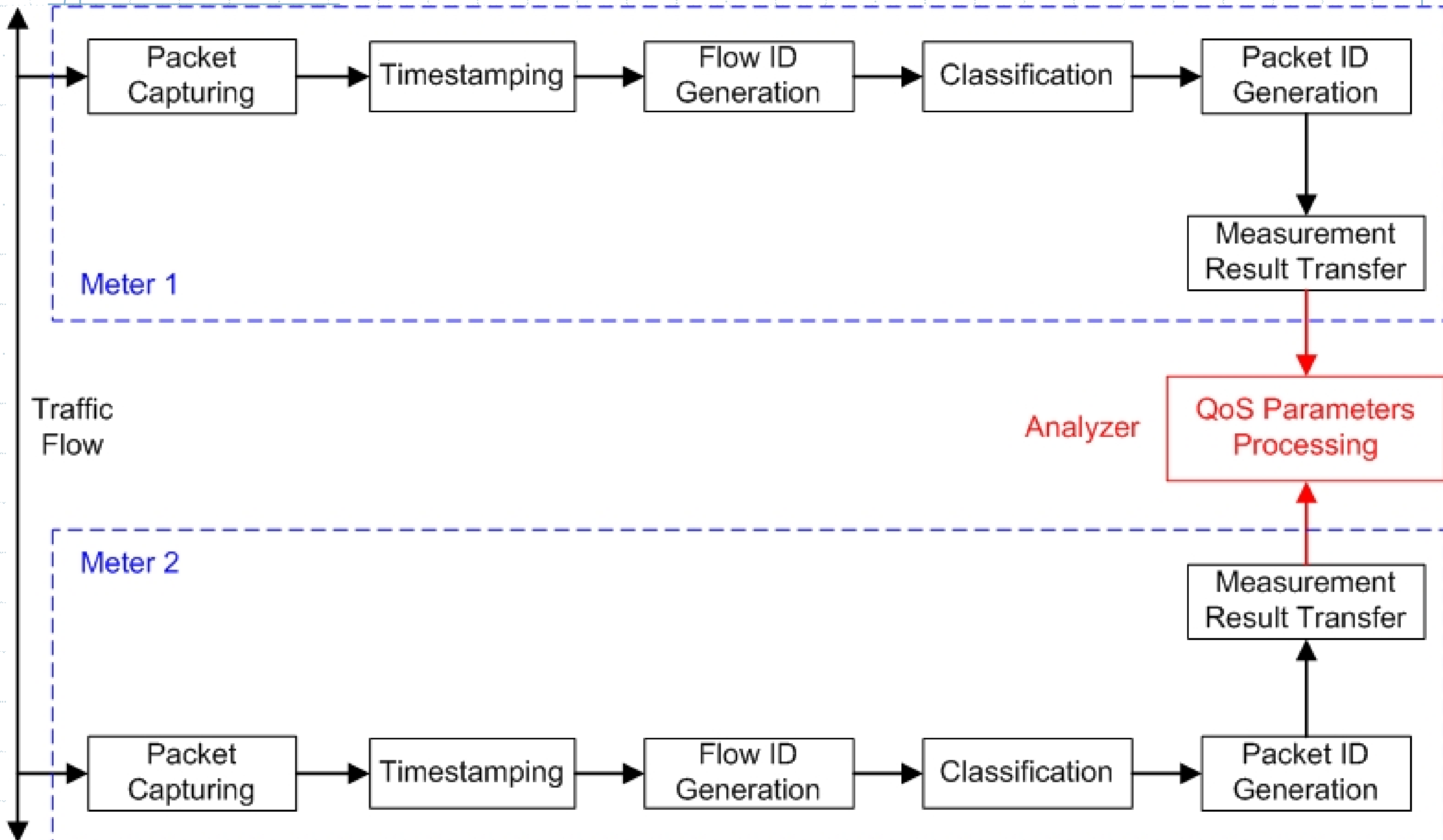
# 3. Designing of the Measurement Tool (I)

- **Flows:** defined by the *five-tuple*
  - IPv4/IPv6 Source Address
  - IPv4/IPv6 Destination Address
  - Source Port
  - Destination Port
  - Transport Layer Protocol

| 0 – 31 | 63 | 79 | 95 | 103 |
|---|---|---|---|---|
| IPv4 Source Address | IPv4 Destination Address | Source Port | Destination Port | Protocol Number |

| 0 – 127 | 255 | 271 | 287 | 295 |
|---|---|---|---|---|
| IPv6 Source Address | IPv6 Destination Address | Source Port | Destination Port | Protocol Number |

# 3. Designing of the Measurement Tool (II)



# 3. Designing of the Measurement Tool (III)

- **Packet capturing**
  - *pcap* library (libpcap 0.8.3 or later)
- **Time-stamping**
  - Capturing points must be synchronized
  - *GPS (Global Positioning System)*
  - *NTP (Network Time Protocol)*
- **Flow ID** – CRC-32 on the *five-tuple*:
- **Classification**
  - Each packet is assigned a sequence number depending on the flow it belongs to
- **Packet header** fields
  - Must not vary during the transport through the network
  - Should be highly variable between different packets
- **Data** fields
  - first 27 bytes of payload

# 3. Designing of the Measurement Tool (IV)

## Binary File

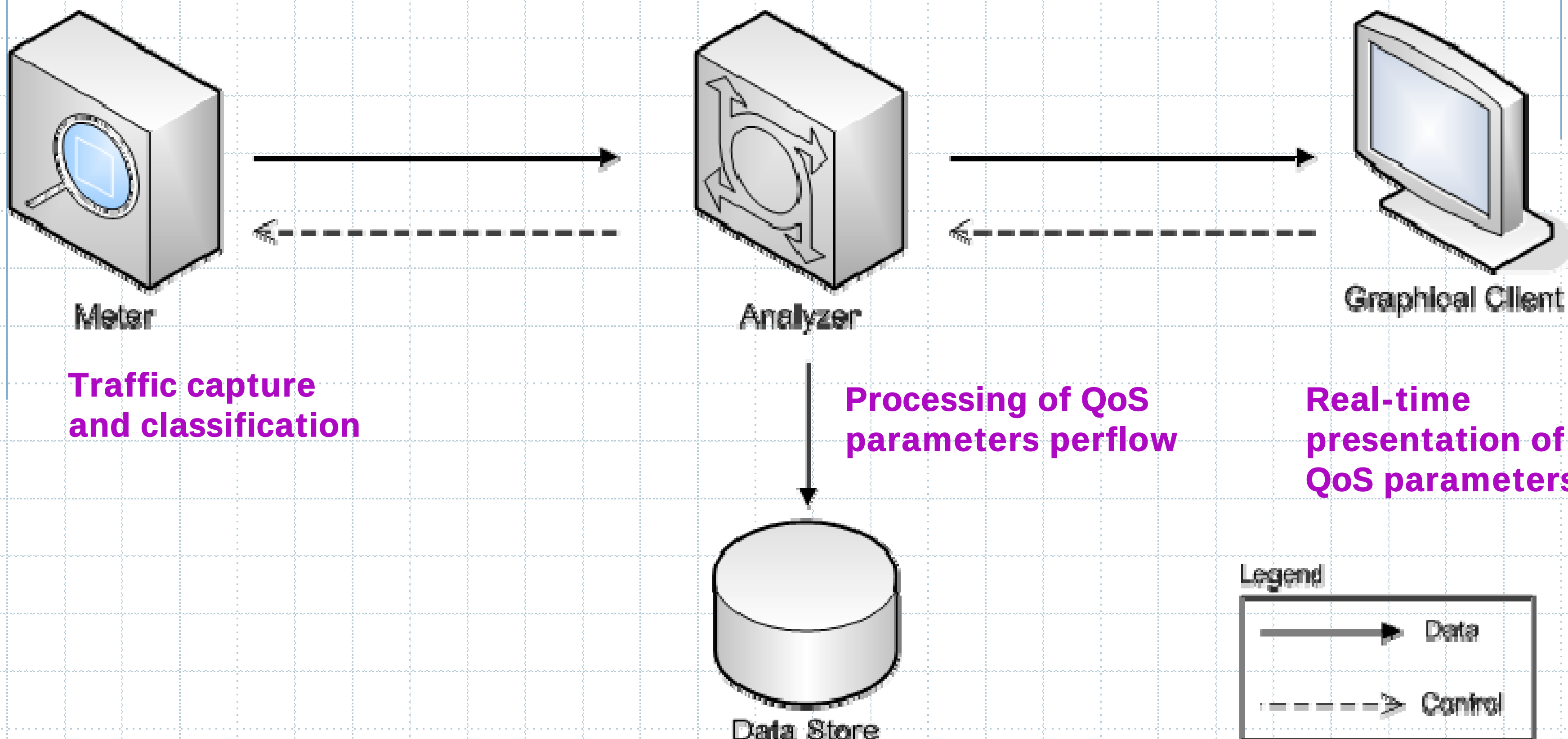
```
        crc_packet          seq nr          sec             usec
        FC DC 84 31 01 00 00 00|42 93 67 7A 00 0E 20 ED | ìÜ■1■■■■B■gz■■ í
size <- 03 00 47 D7 9B D6 02 00|00 00 42 93 67 7A 00 0E | ■■G×■Ö■■■■B■gz■■
        6F 0B 03 00 F6 BE 03 91|03 00 00 00 42 93 67 7A | o■■■ö¾■‘■■■■B■gz
        00 0E BD 2C 03 00 46 28|9A 92 04 00 00 00 42 93 | ■■½,■■F(■’■■■■B■
        67 7A 00 0F 0B 4B 03 00|6B A0 28 E8 05 00 00 00 | gz■■■K■■k (è■■■■
        42 93 67 7B 00 00 17 28|03 00 68 99 5C 24 06 00 | B■g{■■■(■■h■\$■■
        00 00 42 93 67 7B 00 00|65 53 03 00 B8 83 81 F9 | ■■B■g{■■eS■■¸■■ù
        07 00 00 00 42 93 67 7B|00 00 B3 62 03 00 D1 EF | ■■■■B■g{■■³b■■Ñï
        49 F3 08 00 00 00 42 93|67 7B 00 01 01 82 03 00 | Ió■■■■B■g{■■■■■■
        84 52 12 8C 09 00 00 00|42 93 67 7B 00 01 4F 9E | ■R■■■■■■B■g{■■O■
        03 00 64 31 57 75 0A 00|00 00 42 93 67 7B 00 01 | ■■d1Wu■■■■B■g{■■
        9D BE 03 00 77 73 91 A9|0B 00 00 00 42 93 67 7B | ■¾■■ws‘©■■■■B■g{
        00 01 EB DB 03 00 F0 06|C8 52 0C 00 00 00 42 93 | ■■ëÛ■■ð■ÈR■■■■B■
        67 7B 00 02 39 FB 03 00|19 39 E1 1C 0D 00 00 00 | g{■■9û■■■9á■■■■■
        42 93 67 7B 00 02 88 16|03 00 6F 54 55 B6 0E 00 | B■g{■■■■■■oTU¶■■
        00 00 42 93 67 7B 00 02|D6 33 03 00 07 67 9E 16 | ■■B■g{■■Ö3■■■g■■
        0F 00 00 00 42 93 67 7B|00 03 24 61 03 00 F8 68 | ■■■■B■g{■■$a■■øh
```

# 3. Designing of the Measurement Tool (V)

## Aggregate File

```
Flow>0001 Seq>000000 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:18.925933 RxTime>19:42:18.926814 Size>0768
Flow>0001 Seq>000001 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:18.945931 RxTime>19:42:18.946811 Size>0768
Flow>0001 Seq>000002 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:18.965932 RxTime>19:42:18.966815 Size>0768
Flow>0001 Seq>000003 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:18.985931 RxTime>19:42:18.986818 Size>0768
Flow>0001 Seq>000004 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:19.005928 RxTime>19:42:19.006814 Size>0768
Flow>0001 Seq>000005 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:19.025939 RxTime>19:42:19.026828 Size>0768
Flow>0001 Seq>000006 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:19.045922 RxTime>19:42:19.046810 Size>0768
Flow>0001 Seq>000007 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:19.065922 RxTime>19:42:19.066812 Size>0768
Flow>0001 Seq>000008 Src>  192.168.11.3/1306  Dest>  192.168.11.4/5000  TxTime>19:42:19.085918 RxTime>19:42:19.086810 Size>0768
```

# 4. Proposed Architecture (I)




Traffic capture and classification

Processing of QoS parameters perflow

Real-time presentation of QoS parameters

**Passive measurement**

# 4. Proposed Architecture (II)

## OReNETa's Packet Reports

- IPv4: 14-byte redundancy (14+14=28 bytes)

| 0 | 1 | 5 | 9 | 13 | 15 | 19 | 23 | 24 | 26 | 28 |
|---|---|---|---|---|---|---|---|---|---|---|
| L3 Protocol | sec | usec | crc | IPv4 Length | IPv4 Src Address | IPv4 Dst Address | L4 Protocol | L4 Src Port | L4 Dst Port | |

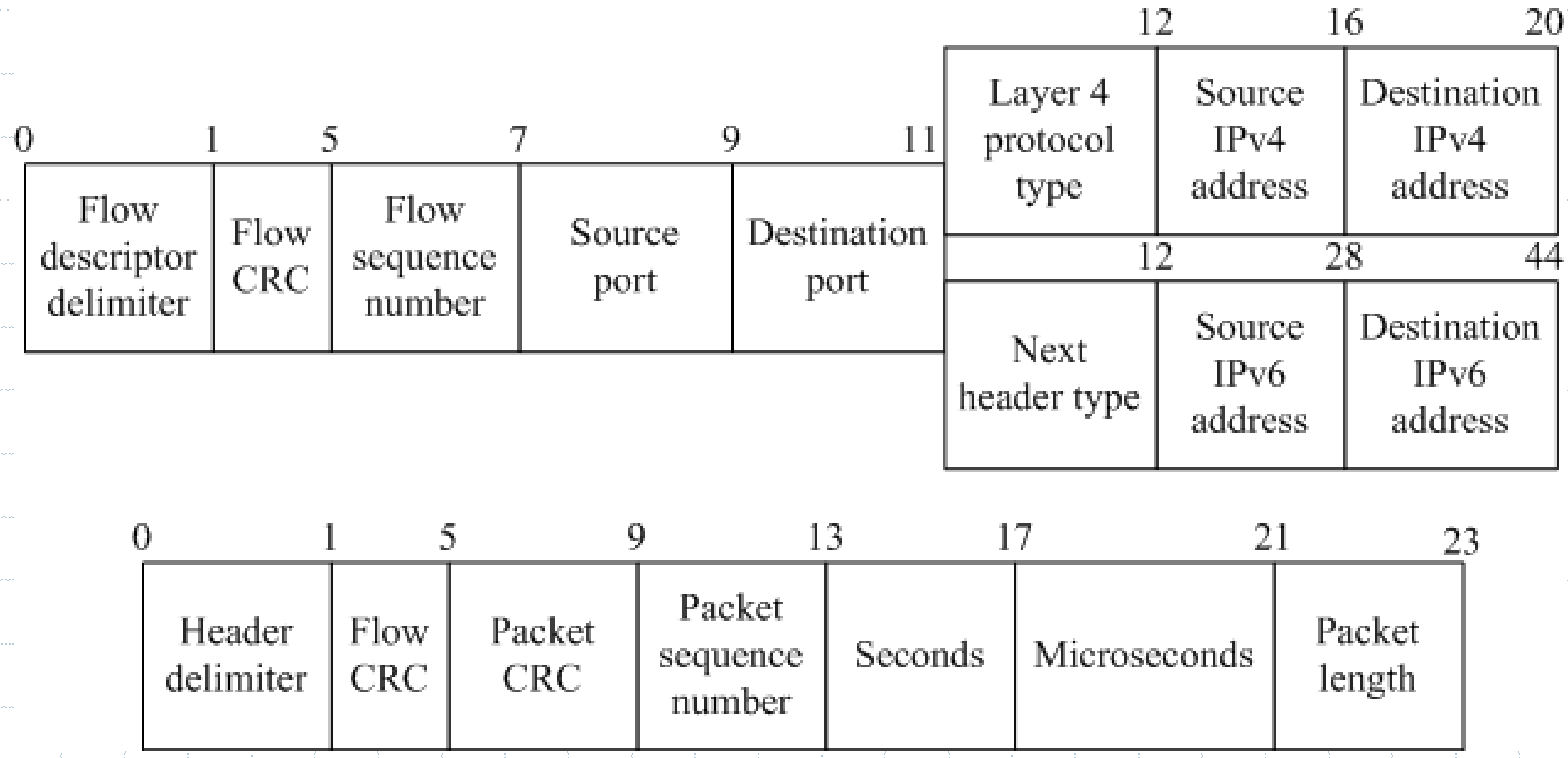
- IPv6: 38-byte redundancy (14+38=52 bytes)

| 0 | 1 | 5 | 9 | 13 | 15 | 31 | 47 | 48 | 50 | 52 |
|---|---|---|---|---|---|---|---|---|---|---|
| L3 Protocol | sec | usec | crc | IPv6 Length | IPv6 Src Address | IPv6 Dst Address | L4 Protocol | L4 Src Port | L4 Dst Port | |

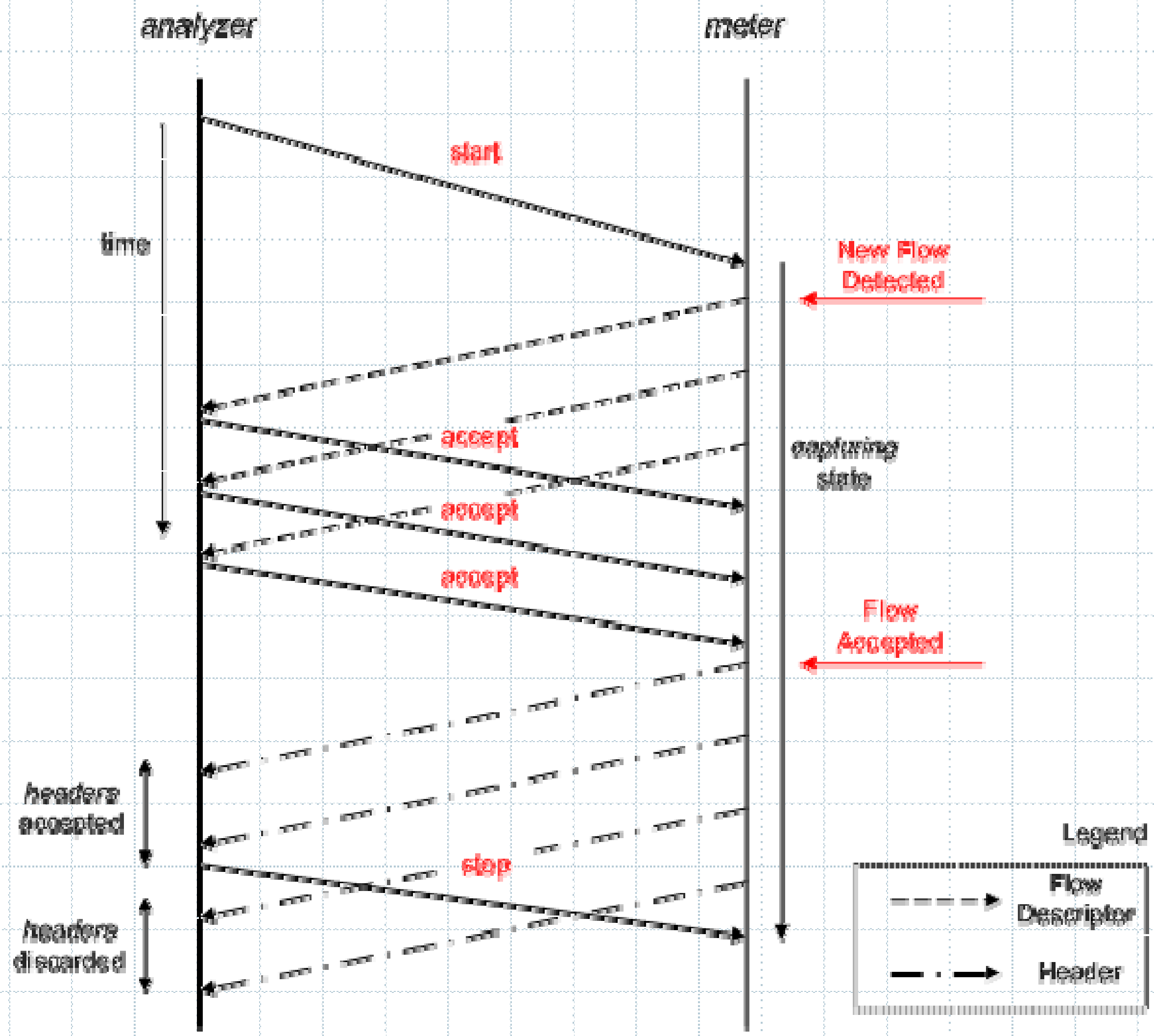
# 4. Proposed Architecture (III)

## New Approach

- Flow Descriptors (sent at the beginning of capture)
- Headers (23 bytes instead of 28/52 bytes)



# 4. Proposed Architecture (IV)

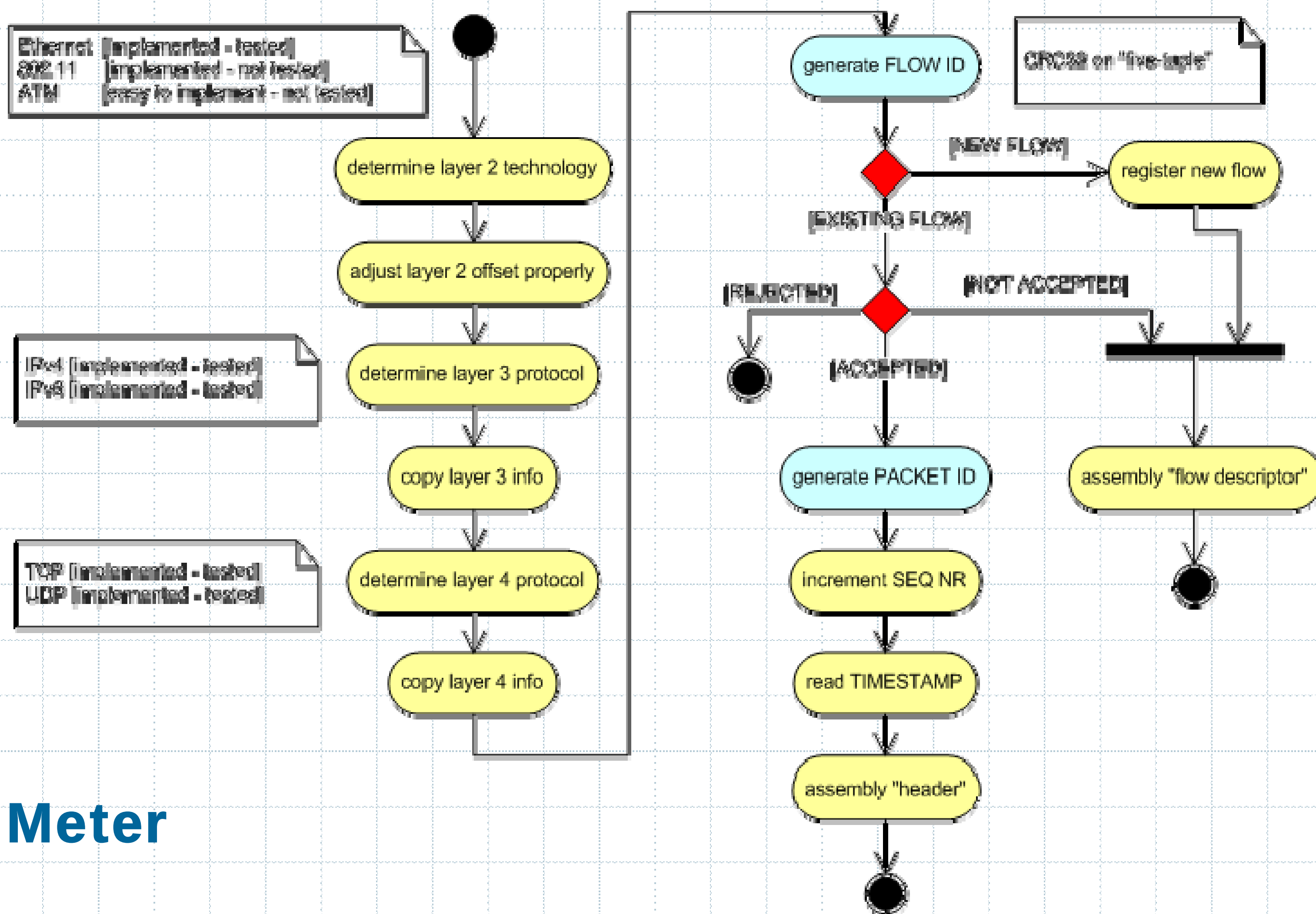


# 4. Proposed Architecture (V)

## Meter:

- Packet Capturing (use *libpcap*)
- Timestamping (use *libpcap*)
- Flow ID Generation (CRC-32 of the five-tuple)
- Classification (generate the binary tree of flows – and send *flow descriptor* or *header*)
- Packet ID Generation (compute CRC-32 for highly variable data in the packet)
- Send the measurement results to the *Analyzer*

# 4. Proposed Architecture (VI)



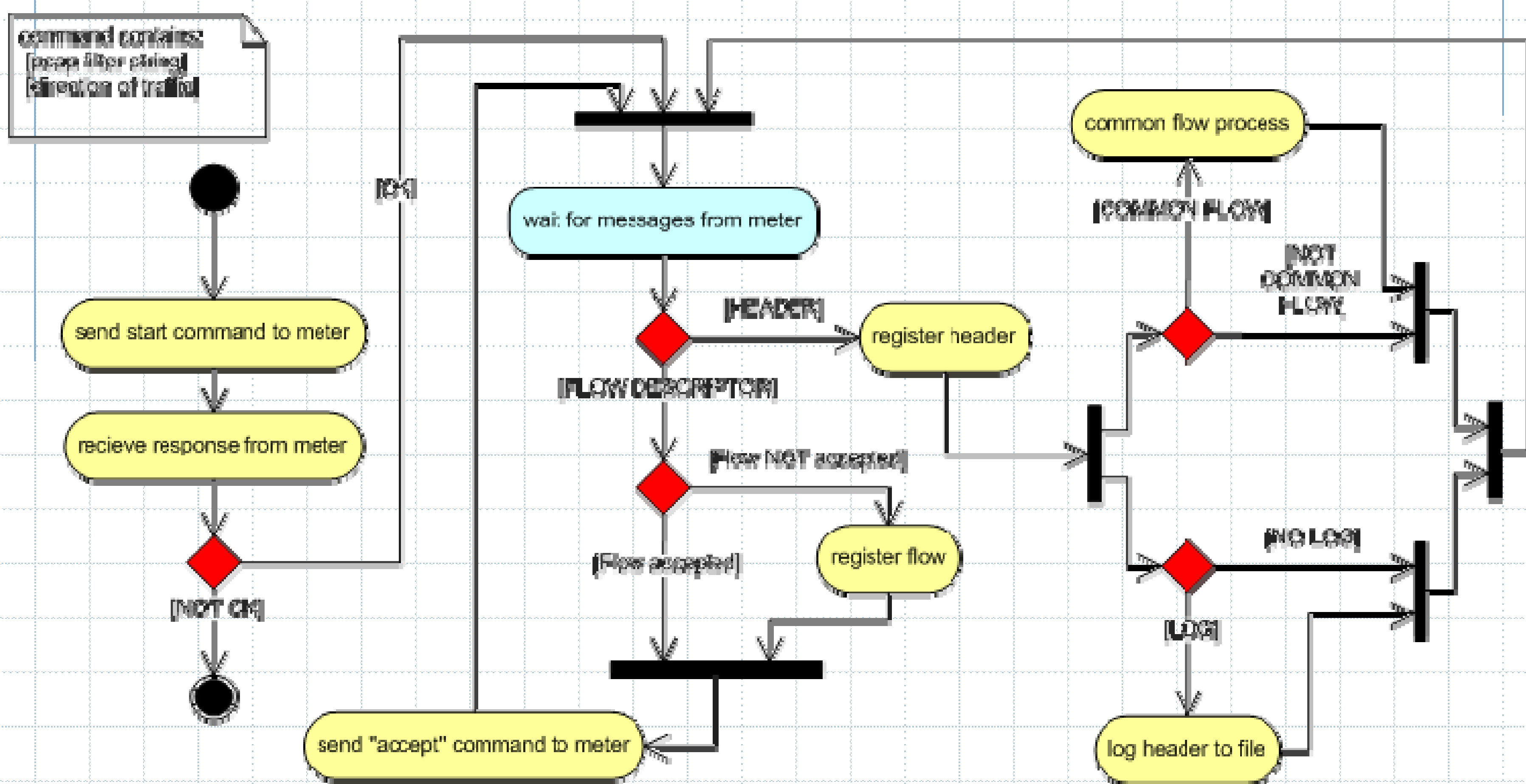

## Meter

# 4. Proposed Architecture (VII)

## Analyzer:

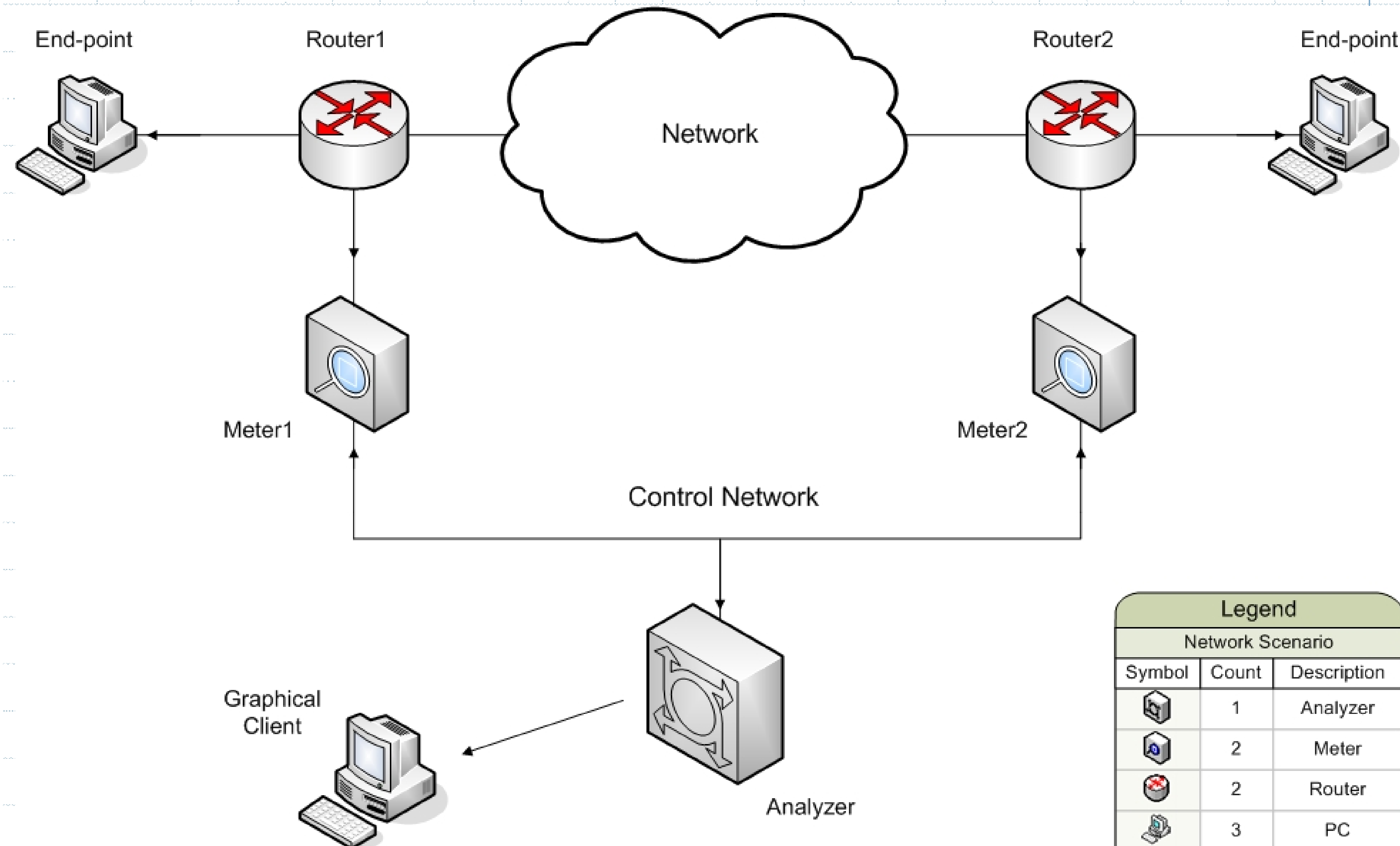
- Connect to the *meters* and command them through the communication protocol
- Fetch the messages from the *meters* (*flow descriptors* or *headers*) and handle them
- Route the *headers* to the appropriate flow structures (binary tree)
- Synchronize the received *headers* to compute the QoS parameters
- Present the data to Graphical clients
- Handle interactive commands from the command-line

# 4. Proposed Architecture (VIII)

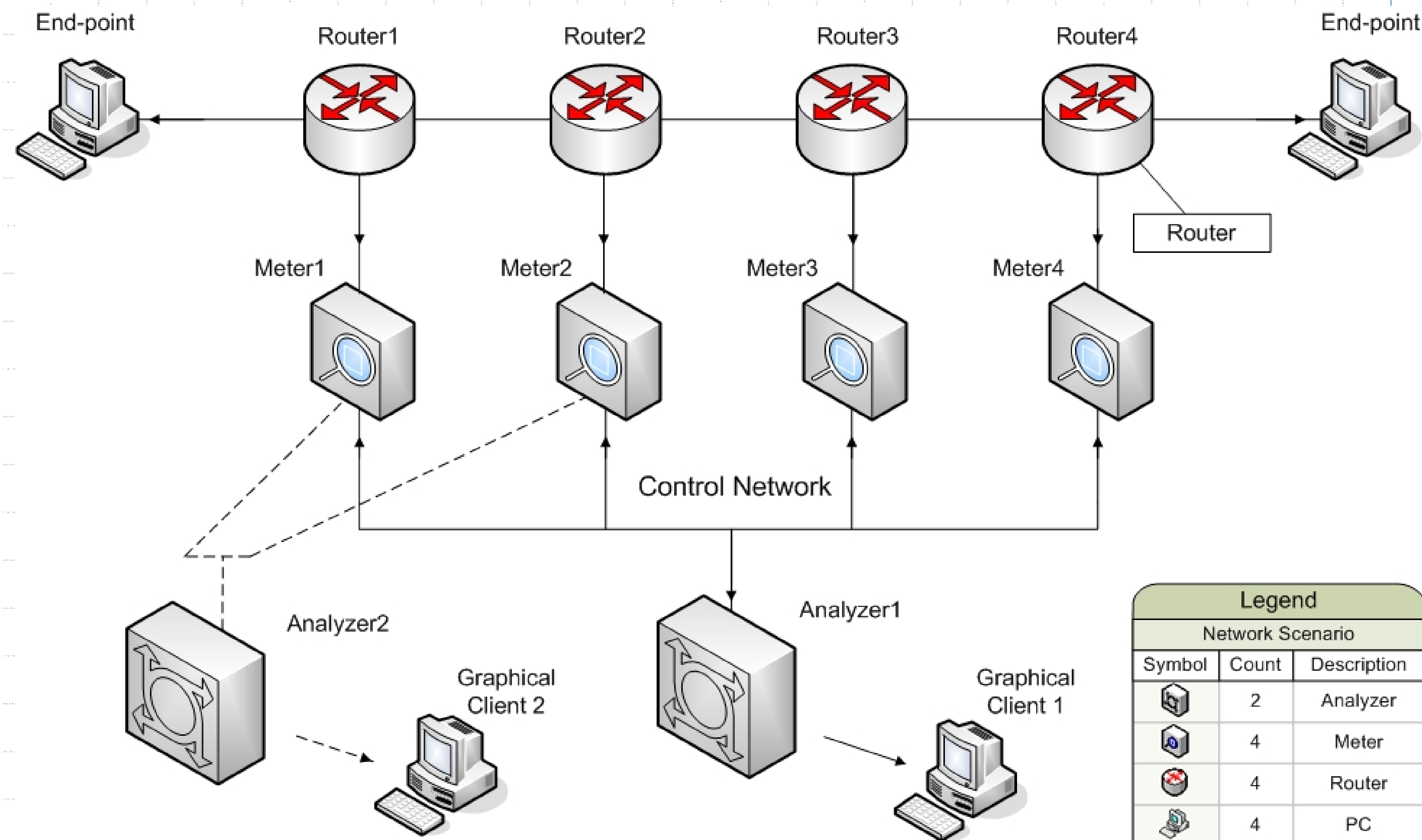



## Analyzer

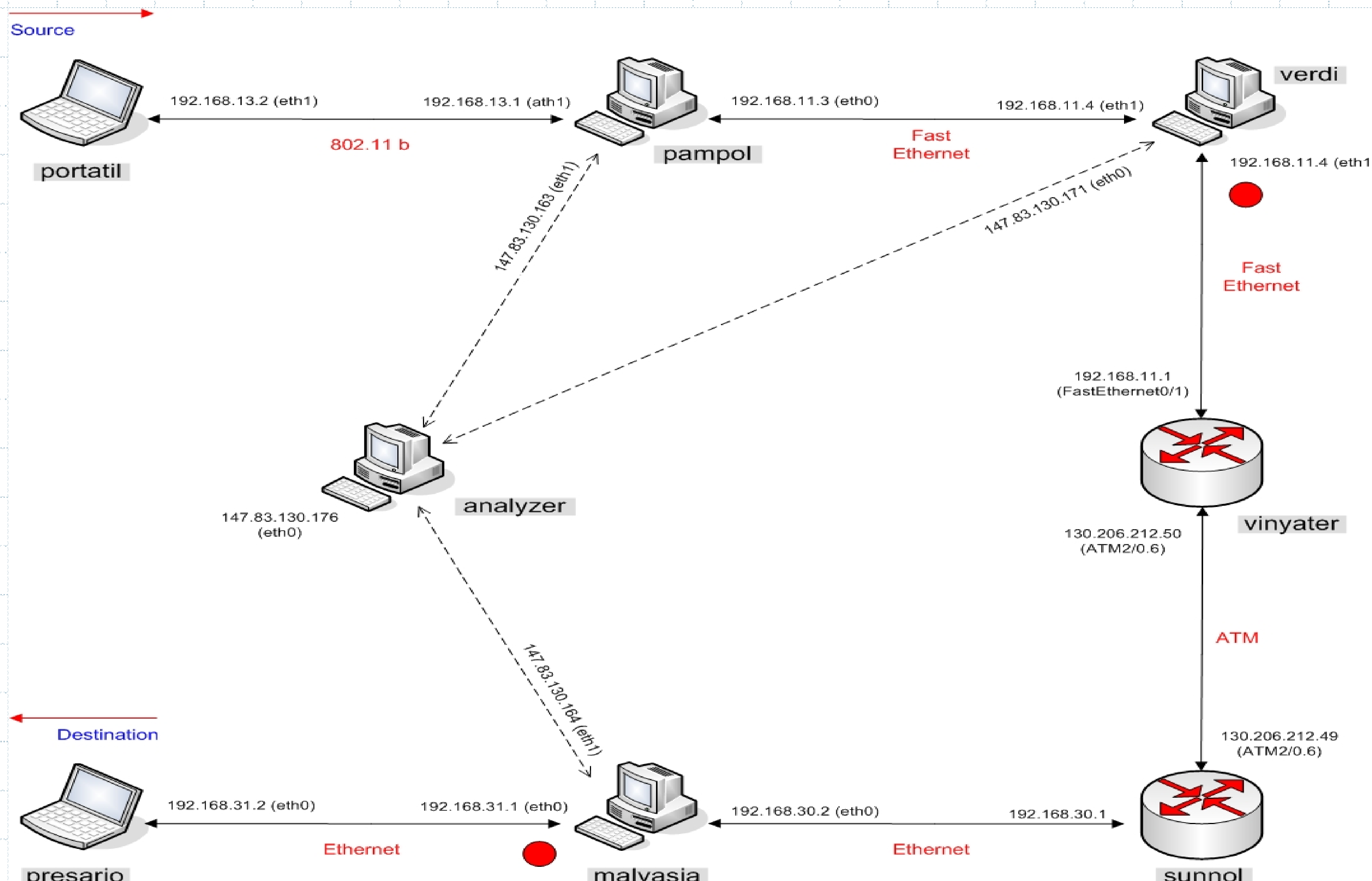
# Basic Architecture



# Complex Architecture



# Testbed – 2 Capture Points



# Offline Results from 2 *meters*

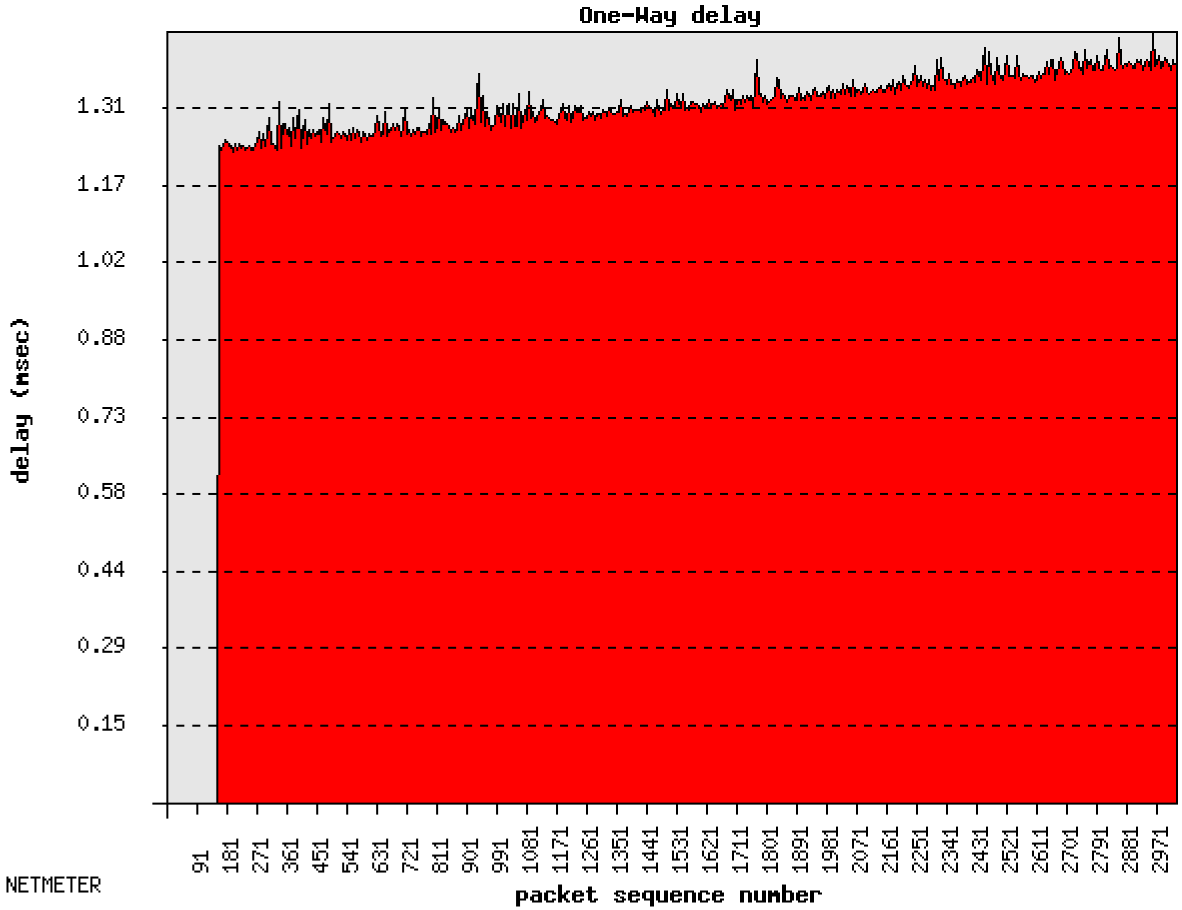
- 30 seconds MGEN traffic from *Portatil* to *Presario*, with **meters** running on *Verdi* and *Malvasia* :

```
mgen -i eth0 -b 192.168.31.2 -s 370 2000000 -r
100
```

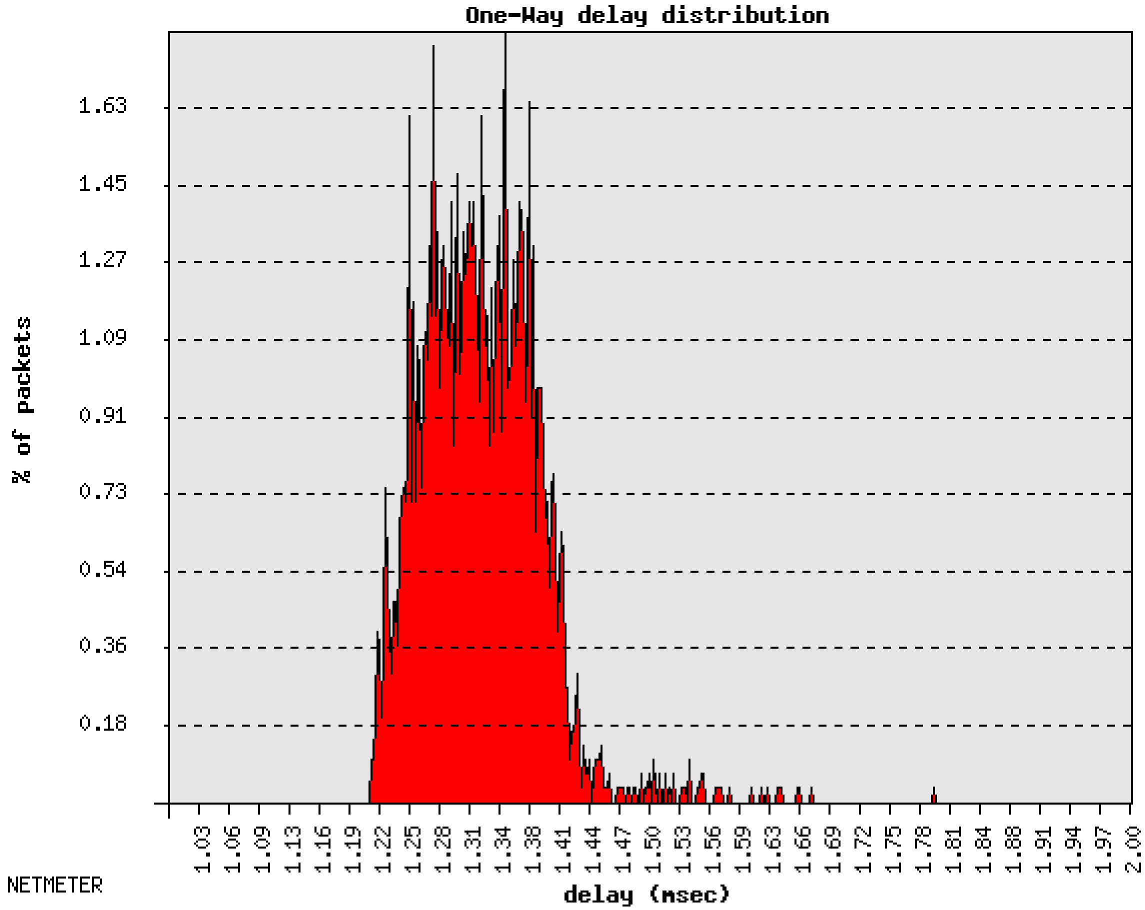
- **Analyzer** command-line:

```
./analyzer -l
  -i 147.83.130.164 1   -d eth0 1   -w out 1
  -i 147.83.130.171 2   -d eth1 2   -w out 2
  -f "udp"
```

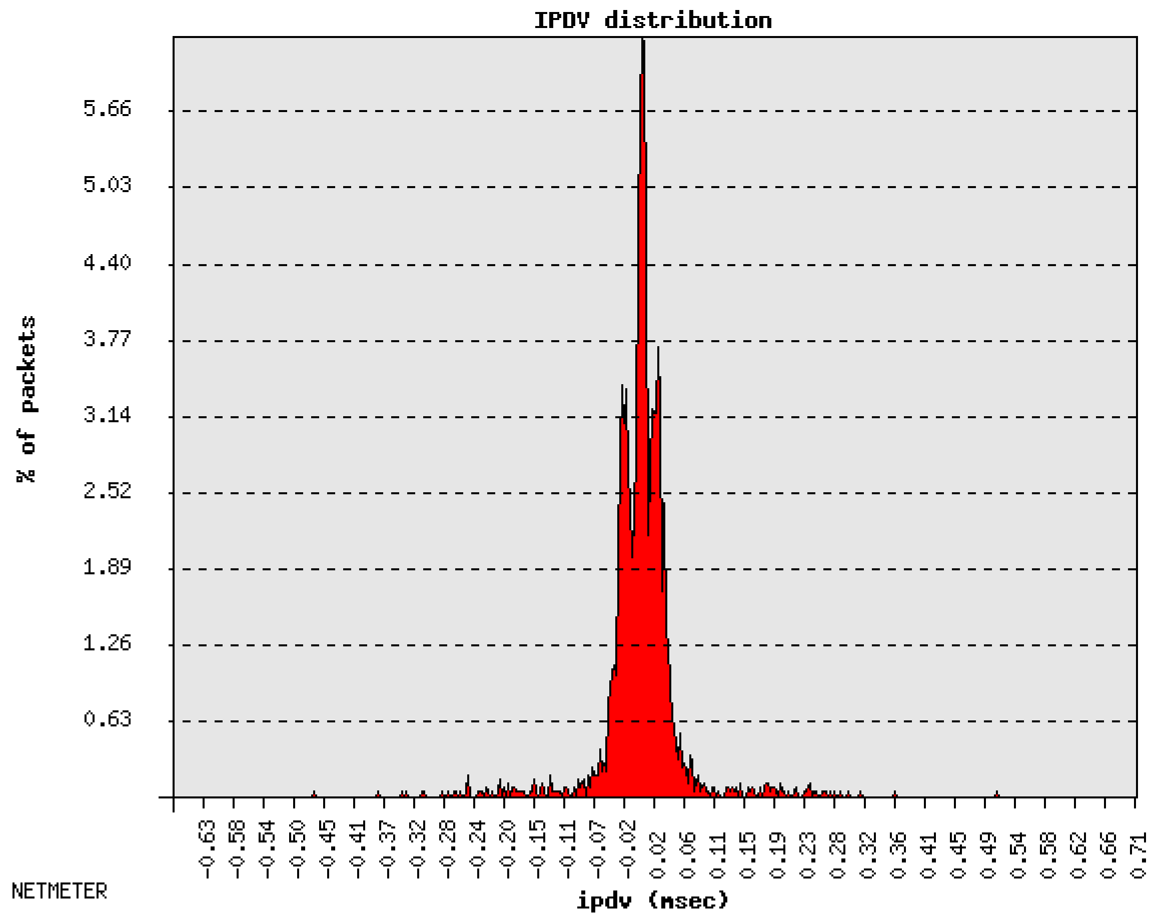
# OWD for 3000 packets



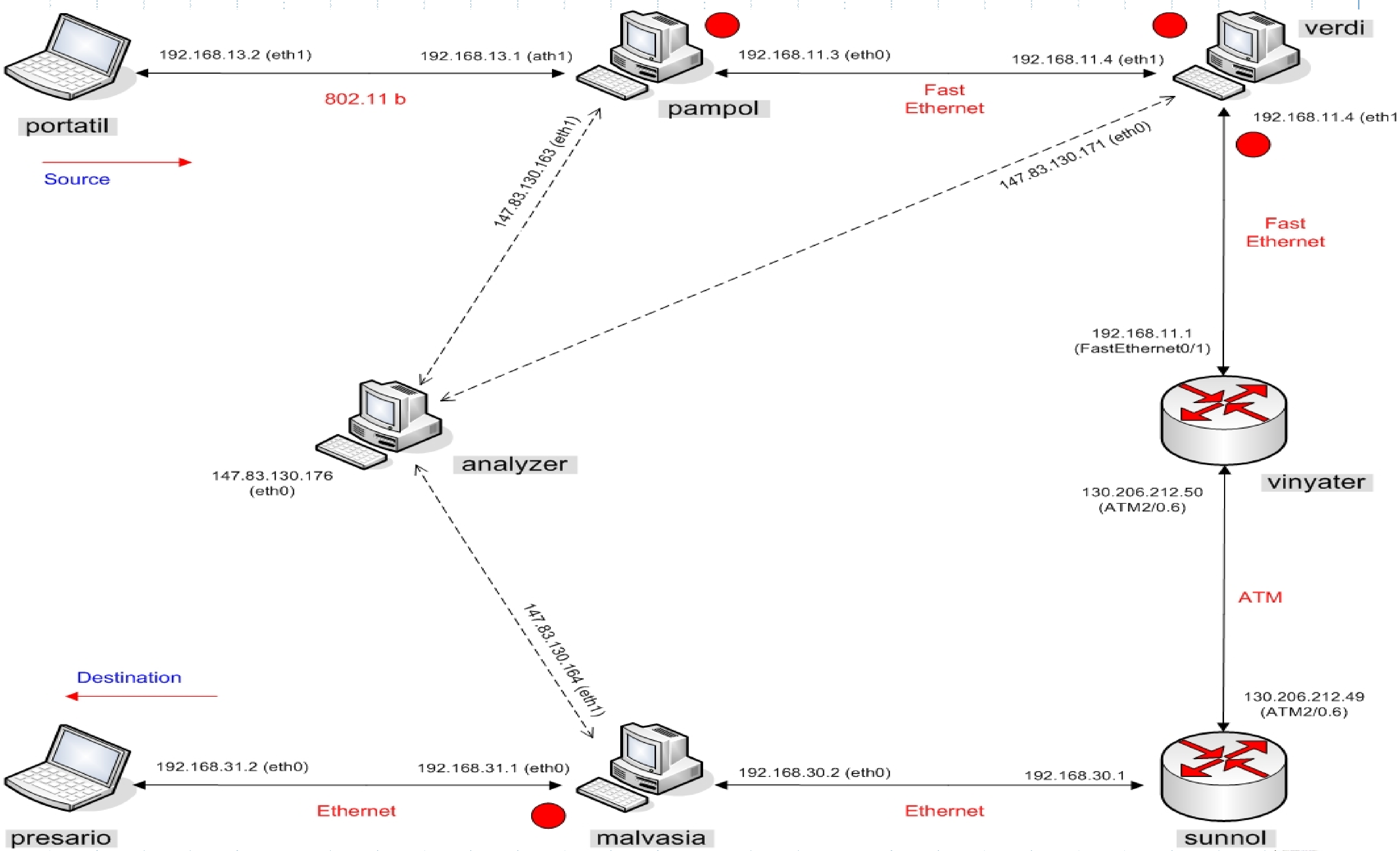
# OWD Distribution for 3000 packets



# IPDV Distribution for 3000 packets



# Testbed – 4 Capture Points



# Real-time Testing with 4 *meters*

- MGEN traffic from *Portatil* to *Presario*, with **meters** running on *Pampol*, *Verdi* (in and out) and *Malvasia* :

```
mgen -i eth0 -b 192.168.31.2 -s 512 2000000 -r 200
```

- **Analyzer** command-line:

```
./analyzer
  -i 147.83.130.163 1   -d eth0 1   -w out 1
  -i 147.83.130.171 2   -d eth1 2   -w out 2
  -i 147.83.130.164 3   -d eth0 3   -w out 3
  -i 147.83.130.171 4   -d eth1 4   -w in  4
  -f "udp and host 192.168.31.2"
```

# Results – Real-time Screen Capture

| MTR | M[1] | M[4] | M[2] | M[3] |
|---|---|---|---|---|
| PPS | 200 packets/s | 200 packets/s | 200 packets/s | |
| THROUGHPUT | 108000 bytes/s | 108000 bytes/s | 108000 bytes/s | |
| OWD | 0.001656 seconds | 0.000210 seconds | 0.007587 seconds | |
| IPDV | 0.000000 seconds | -0.000000 seconds | 0.000000 seconds | |
| PKLOSS | 0 packets/s | 0 packets/s | 0 packets/s | |

# Conclusions

- Improvements, optimizations and additional functionalities compared to original OReNETa
- *flow descriptors* and *headers* optimize the traffic between the *meter* and the *analyzer* . The values identifying a flow (*five-tuple*) are only sent with the *flow descriptor*.
- Size of packet reports: reduced from 28 to 23 bytes (IPv4), and from 52 to 23 bytes (IPv6).
- Irrelevant packets (e.g. ICMP and ARP) were discarded at the meter
- More than one *analyzer* to a *meter* and vice-versa
- Pure binary files to store only the needed data, and the processing is performed later, when the capture is finished.