



# NetDisturb

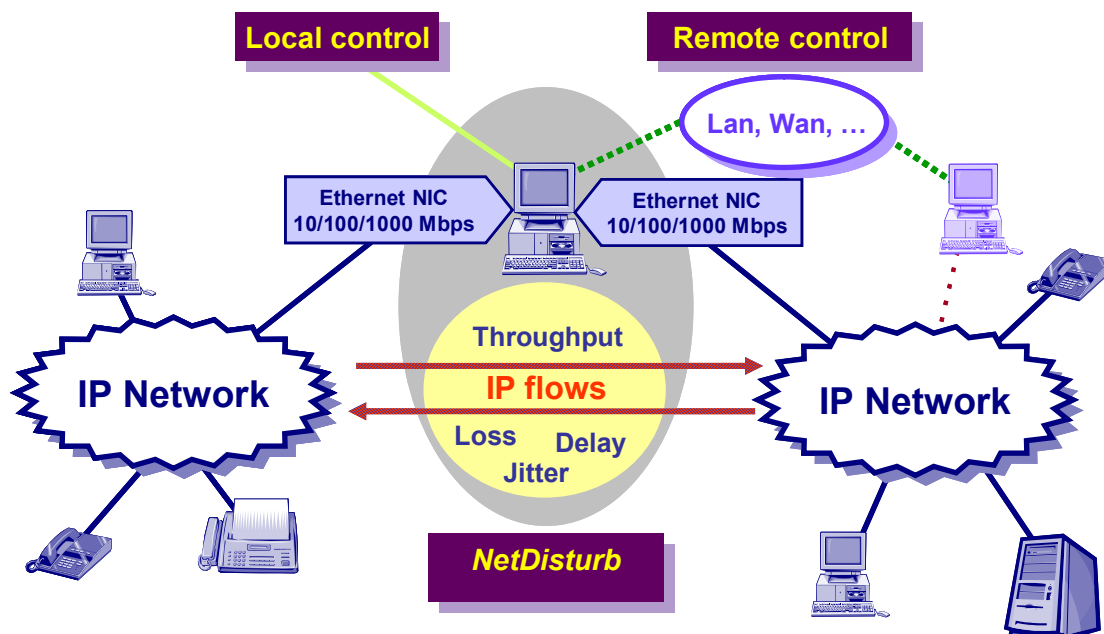
Version 4.4

## *Impairment Emulator Software for IP Networks (IPv4 & IPv6)*

### Product Overview

**NetDisturb** is an IP network emulator software that can generate impairments (latency, delay, jitter, bandwidth limitation, lost and duplicate packets) over IP networks (IPv4 and IPv6). **NetDisturb** allows the user to disturb flows on an IP network and so to study the behavior of applications, devices or services in a disturbed network environment.

**NetDisturb** is inserted between two Ethernet segments (on the same IP network or two different IP networks) and operates bi-directional packet transfer on Ethernet, Fast Ethernet and Gigabit network interface cards.



### Product Requirements

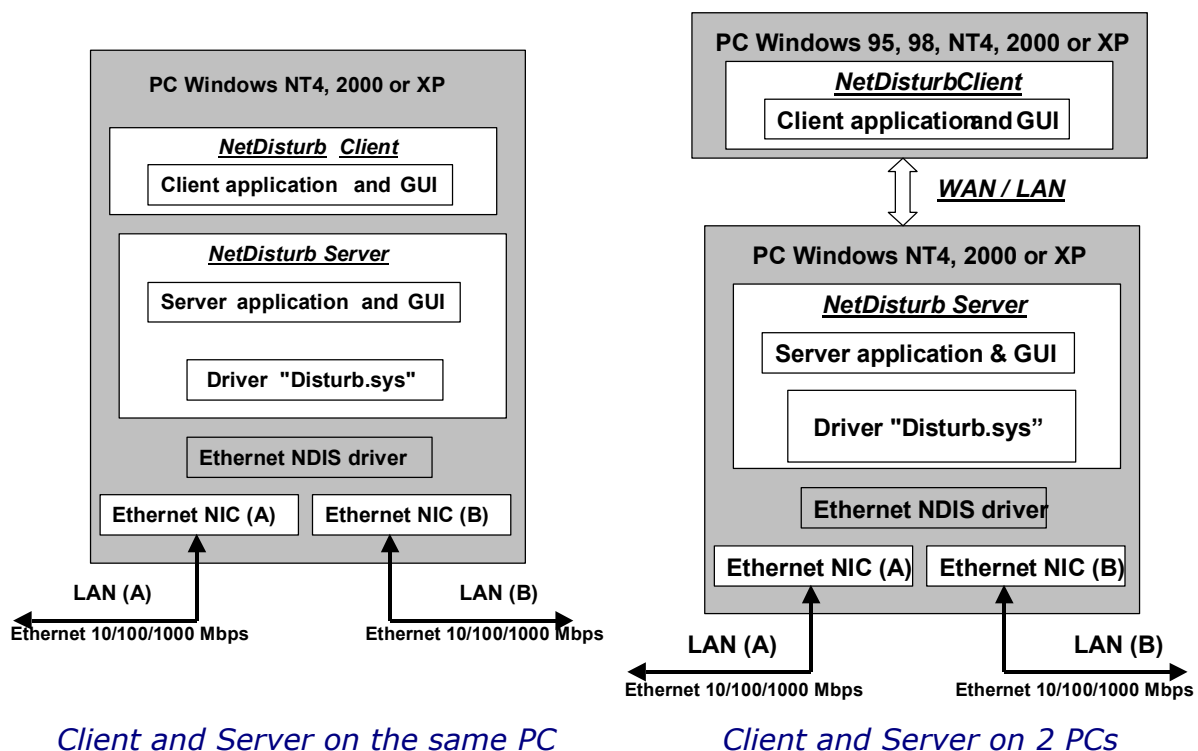
- \* Platform: pentium PC running Windows NT4 (SP6), 2000 or XP with Microsoft TCP/IP installed and at least 256 MB Ram.
- \* Hyper-threading and PC multiprocessors are also supported.
- \* Two Identical Network Interfaces Cards (NIC): Ethernet, Fast Ethernet, or Gigabit Ethernet network interface card.
- \* 1024 x 768 display and DPI setting = Normal size (96 DPI).

### Configurations

Based on a Client-Server architecture, the **NetDisturb** software is made of two parts: Server and Client. The Server handles the impairment characteristics and the Client manages the Server using a simple graphical interface.

This allows two configurations where the Server and the Client parts may be installed on the same PC host (local control), or the Server part is located on one PC and the Client part is located on a second PC (remote control). In this second configuration, the Client dialogs with the Server by using a Wan (for example: PSTN or ISDN) or a LAN link.

Both configurations require two identical Ethernet Cards for the Server.



The "Disturb.sys" driver is located in the kernel of the operating system and is installed above the NIC drivers. This driver is used by **NetDisturb** to handle the exchanges with the NICs.

### Products features

What are the major features of **NetDisturb** V4.4?

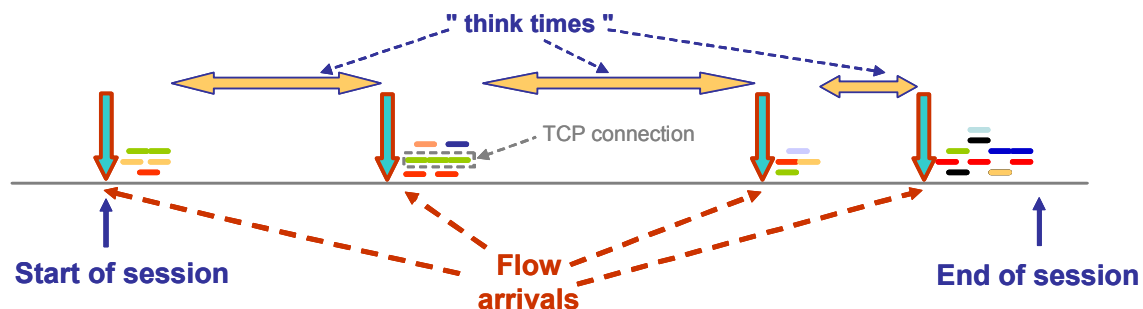
#### Key features

- Client-Server Architecture
- Impairments: Latency, Loss, Duplication, bandwidth limitation, Delay and Jitter (mathematical laws and user-defined files)
- 16 configurable IP flows per direction with optional trigger condition
- Aggregates of IP flows can be defined (set of IP flows sharing the same Delay & Jitter Law)
- Unidirectional or bidirectional packet impairments
- Connections per IP flow: impairments are applied to the IP flow or to each connection of the IP flow
- Ethernet / Internet modes (desequencing of the packets)
- Easy to use and intuitive Graphical User Interface
- Statistics display and export detailed statistics in a file

**NetDisturb** is based on the notion of IP flows.

A flow is a set of packets with a set of common packet properties, and can be unidirectional or bidirectional.

Flows are part of sessions (successions of flows and "think times") related to some homogeneous user activity (e-commerce, mail, MP3 file, web, etc.).



An IP flow is described by using a n-tuple.

In the typical case, the following 5-tuple is used: IP addresses, protocol and port numbers.

An IP flow is composed of connections (such as TCP connections to make FTP transfer by example).

To define the n-tuple for an IP flow, **NetDisturb** uses the notion of mask.

A mask is the combination of the following optional parameters:

**IP version (IPv4, IPv6 or IPv4 & IPv6)**

**Ethernet header**

- MAC destination address
- MAC source address

**List of VLAN-ID (Ethernet frames 802.1Q)**

### IP Header

- Destination IP address
- Source IP address
- Protocol (ICMP, TCP, UDP, SIP, ...)
- Differentiated services (TOS)

### List of Ports (for TCP or UDP packets)

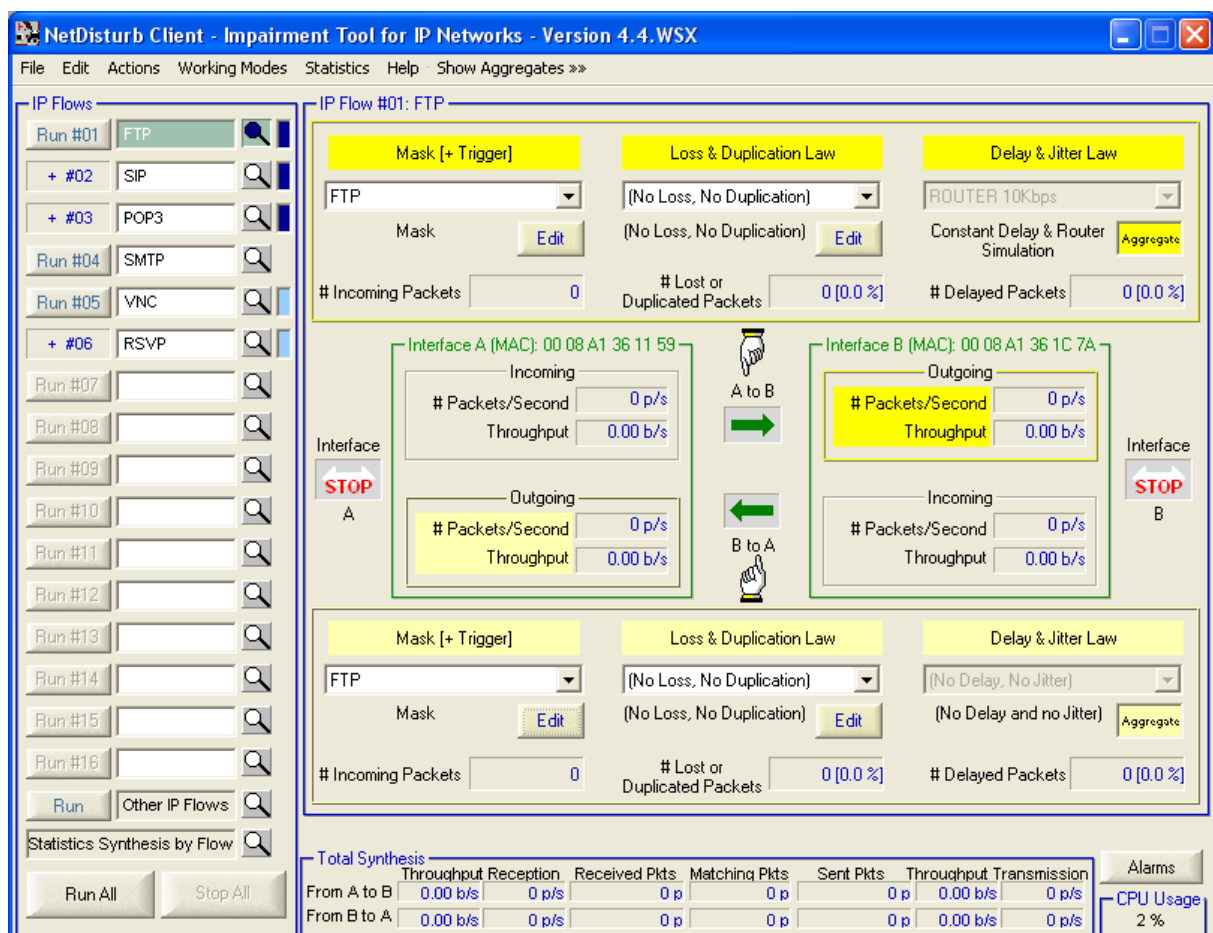
- Destination port list
- Source port list

*Note: a trigger can be associated optionally with the mask.*

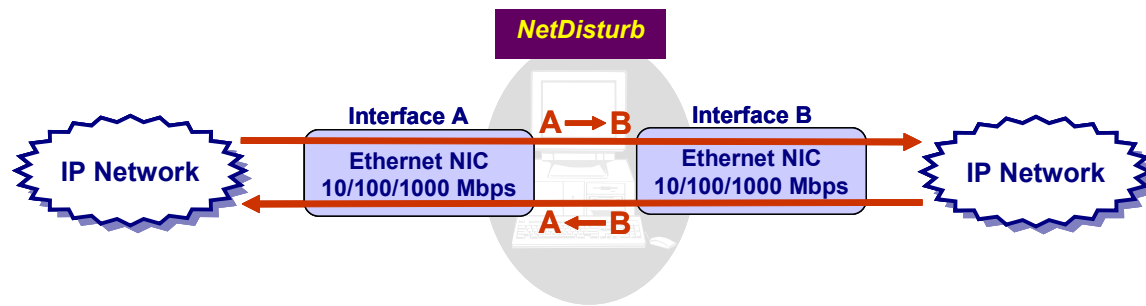
With **NetDisturb** you can define up to 16 masks, i.e. 16 IP flows. An additional item named "Other IP Flows" is in charge to handle all IP flows that have not been user defined. For this item no mask can be defined, but impairments can be applied.

**NetDisturb** manages up to 10 000 connections – all flows included.

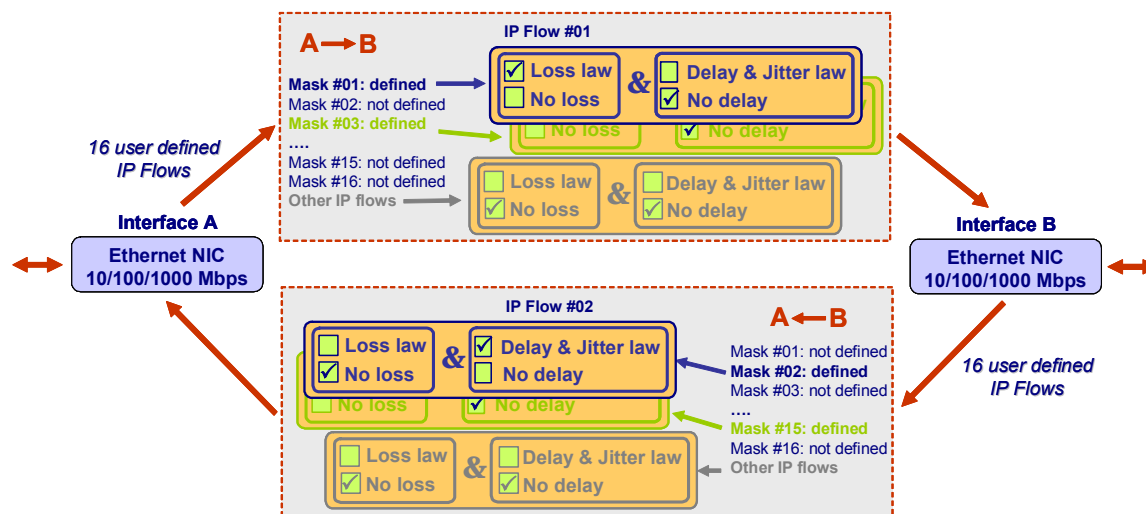
The client window below illustrates the management of IP flows by **NetDisturb**.



The graphical user interface represents the NIC cards as "Interface A" and "Interface B" as illustrated below.



For each direction  $A \rightarrow B$  or  $B \rightarrow A$ , 16 flows can be defined by the user. And for each IP flow, loss & duplication and / or delay laws can be applied as shown in the figure below.



In the above example, **NetDisturb** has been configured with the following parameters:

## Direction $A \rightarrow B$

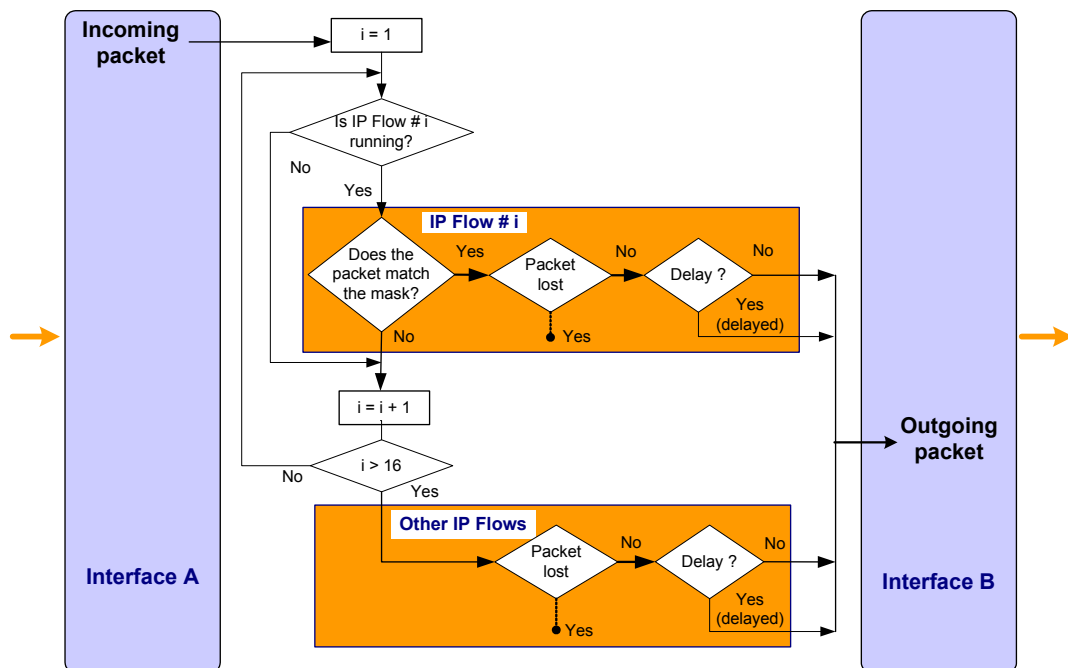
- the Mask #01 defines the "IP Flow #01", and a loss law is applied to the packets of this flow,
- the Mask #03 defines the "IP Flow #03", and a loss law is applied to the packets of this flow,
- As no loss and delay law is applied to the 'Other IP flows', all packets that don't match the masks #01 and #03 are relayed directly from the interface A to B.

## Direction $B \rightarrow A$

- the Mask #02 defines the "IP Flow #02", and a delay law is applied to the packets of this flow,
- the Mask #15 defines the "IP Flow #15", and a loss law is applied to the packets of this flow,
- As no loss and delay law is applied to the 'Other IP flows', all packets that don't match the masks #02 and #15 are relayed directly from the interface B to A.

### How does it work?

We illustrate how **NetDisturb** handles incoming packets with the following scheme from the A interface to the B interface.



Depending on the active user-defined IP flows, **NetDisturb** identifies if the incoming packet belongs to an IP flow before applying loss or delay treatments. If this packet matches with the mask of an IP Flow (IP Flow #i for example), then **NetDisturb** identifies if this packet must be lost/duplicated and/or delayed. If this packet does not match any mask (a mask defines an IP flow), then **NetDisturb** applies the treatments for the 'Other IP Flows' and identifies if this packet must be lost/duplicated and/or delayed. For each packet received on an interface, **NetDisturb** analyzes in order the masks from 1 to 16 before considering this packet to belong to the "Other IP Flows".

So **NetDisturb** can apply impairments on the IP flows defined by the user either unidirectional ( $A \rightarrow B$  or  $B \rightarrow A$ ) or bidirectional (the same impairments apply for the two directions:  $A \rightarrow B$  and  $B \rightarrow A$ ).

### Introduction of a Trigger for the Mask

**NetDisturb** allows optionally using a trigger to condition the impairment with an event.

The Trigger is an intermediate step after the frame has been classified into an IP Flow and before the frame is impaired.

The Trigger includes various parameters:

- The **activation condition** based on the Ethernet frame content.
- The **delay before applying the impairments**
- The **impairment duration** (0 = no limit).
- The **number of cycles** for the trigger (0=unlimited) if the impairment duration is not null.

Thus two main categories of triggers are defined:

- the Trigger without limitation of duration to apply the Impairments
- the Trigger with a limitation of duration to apply the Impairments (and optionally a loop counter)

As soon as the activation condition is performed, the impairment on the IP flow can be immediate or delayed with a duration expressed in milliseconds (delay of impairment).

If the impairment is immediate, the frame that has triggered can be included or not (if the delay before impairment is null).

The impairment can be time limited according to a duration expressed in milliseconds.

When **NetDisturb** is running an IP flow with a defined trigger, four states are possible:

- ⇒ **Waiting for the Trigger**: the impairments do not apply. This state is the initial state of the Trigger.
- ⇒ **The Trigger was found**: the impairments still do not apply because a delay is defined before the impairments. This state changes to the next state when the activation condition is reached.
- ⇒ **The Trigger is active**: the impairments are applied.
- ⇒ **The Trigger is finished**: the impairments do not apply any more. This is the final state of the Trigger.

*Note: a Trigger can remain active permanently if no duration limit was defined.*

### *Packet impairments*

#### Pre-defined loss and duplication laws:

- Loss: Constant law  
Parameter: number of packets
- Loss: Uniform law:  $f(x) = dx/(\beta - \alpha)$   
Parameters: alpha, beta, threshold
- Loss: Burst Uniform law:  $f(x) = dx/(\beta - \alpha)$   
Parameters:  $\alpha$ ,  $\beta$ , threshold(n), threshold(n + x), depth
- Loss: User-defined File  
Parameters: file name, threshold
- Loss: Percentage  
Parameter: percentage
- Loss: 1/N (1 packet is lost every N received packets)  
Parameter: range(N)
- Duplication: Percentage (send n times the received packet)  
Parameters: percentage,  $\text{Min} \leq n \leq \text{Max}$
- Duplication: 1/M (duplicate 1 packet n times every M received packets). Parameters: range(M),  $\text{Min} \leq n \leq \text{Max}$
- Duplication: Uniform  $f(x) = dx/(\beta - \alpha)$   
Parameters: alpha, beta, threshold
- Loss (1/N) then Duplication (1/M): the loss law (1/N) is used first before the duplication law (1/M)

#### Pre-defined Delay & Jitter laws:

- Constant Delay & No Jitter  
Parameter = constant delay
- Constant Delay & Exponential Jitter law:  $f(x) = \lambda e^{-\lambda x} dx$   
Parameters: constant delay,  $\lambda$
- Constant Delay & Uniform Jitter law:  $f(x) = dx/(\beta - \alpha)$   
Parameters: constant delay, alpha, beta
- Constant Delay & User File with Jitter values  
Parameters: constant delay, user file
- User File with Delay & Jitter values  
Parameter: user file
- Router Simulation & Constant Delay  
Parameters: IP throughput, max memory, constant delay



- Router Simulation & User File with Delay and Jitter values  
Parameters: IP throughput, max memory, user file
- Constant Delay & User File with Throughput and Duration values  
Parameters: constant delay, user file

### *Working modes*

**NetDisturb** offers two working modes by applying impairments:

- Enable/Disable desequencing of the packets in a flow,
- Impairment laws apply to the IP flow or to each TCP/UDP connection of the IP flow.

These modes are used together.

For example, **NetDisturb** is set with the following modes:

- Enable desequencing of the packets in a flow
- Impairment laws apply to the IP flow

to simulate the Internet network with disturbed flows.

Another example is to use the following modes:

- Disable desequencing of the packets in a IP flow
- Impairment laws apply to each TCP/UDP connection of the IP flow

to disturb VoIP communications in the same way on an Ethernet network.

### **Enable/Disable Desequencing Packets**

Impairment may introduce changes in the packet sequence – for example by introducing different delays for the packets of a flow.

One of the Ethernet characteristics is to keep packets received in order. Internet hasn't got this constraint regarding the packet order: some packets can use one route while others use another one, with the consequence the receiver may get packets unordered.

**NetDisturb** can simulate the Internet network (enable desequencing packets) or can react as Ethernet does (disable desequencing packets).

### **Impairment laws apply to the IP flow or to each TCP/UDP connection of the IP flow**

**NetDisturb** can analyze IP packets to dispatch them into the TCP or UDP connection they belong to. This mode makes possible to apply the same impairment values to each packet of each connection.

For instance if the impairment has been defined with a loss law: lose the third packet for 10 packets received.

- *Impairment laws apply to the IP flow*

When this option is selected, every received packet matching the mask for this flow is considered to belong to the same flow. Processing is carried out in "continue". With the previous example of loss law (lose the 3<sup>rd</sup> packet on 10 received), **NetDisturb** will lose the 3<sup>rd</sup> packet for ten received packets whatever the TCP/UDP connection belongs to.

- *Impairment laws apply to each TCP/UDP connection of the IP flow*

When this option is selected, **NetDisturb** analyses each received packet in order to associate this packet to a TCP or UDP connection already existing by using these parameters: protocol, IP addresses and port numbers. If the connection doesn't exist, a new one is created.

With the previous example of loss law (lose the 3<sup>rd</sup> packet on 10 received), **NetDisturb** will lose the 3<sup>rd</sup> packet for ten received packets of each TCP or UDP connection.

Up to 10000 connections can be handled simultaneously by **NetDisturb**.

### IP Flows and Aggregates

Up to 8 aggregates of IP flows can be defined. An aggregate is a consecutive set of IP flows sharing the same Delay & Jitter Laws. All IP flows of an aggregate share only one aggregate's Delay & Jitter law (with one law per direction).

The IP flow order in the aggregate defines the priority of packets to delay. While the top IP flow packets get the highest priority, the other IP flow packets are queuing until there are no higher priority packets.

In the example illustrated below, two aggregates have been defined:

- the dark blue colored aggregate collects three IP flows (#01, #02 et #03)
- and the light blue aggregate collects the IP flows #04 et #05.

**NetDisturb Client - Impairment Tool for IP Networks - Version 4.4.WSX**

File Edit Actions Working Modes Statistics Help << Hide Aggregates

**IP Flows**

- Run #01 FTP
- + #02 SIP
- + #03 POP3
- Run #04 SMTP
- Run #05 VNC
- + #06 RSVP
- Run #07
- Run #08
- Run #09
- Run #10
- Run #11
- Run #12
- Run #13
- Run #14
- Run #15
- Run #16
- Run
- Other IP Flows
- Statistics Synthesis by Flow
- Run All
- Stop All

**IP Flow #06: RSVP**

**Mask (+ Trigger)**

Mask: RSVP

# Incoming Packets: 0

**Loss & Duplication Law**

(No Loss, No Duplication)

# Lost or Duplicated Packets: 0 [0.0 %]

**Delay & Jitter Law**

ROUTER 10Kbps

Constant Delay & Router Simulation

# Delayed Packets: 0 [0.0 %]

**Interface A (MAC: 00 08 A1 36 11 59)**

Incoming

# Packets/Second: 0 p/s

Throughput: 0.00 b/s

Outgoing

# Packets/Second: 0 p/s

Throughput: 0.00 b/s

**Interface B (MAC: 00 08 A1 36 1C 7A)**

Outgoing

# Packets/Second: 0 p/s

Throughput: 0.00 b/s

Incoming

# Packets/Second: 0 p/s

Throughput: 0.00 b/s

**Aggregates**

- #01
- #02
- #03
- (None) #04
- (None) #05
- (None) #06
- (None) #07
- (None) #08
- (None) #09
- (None) #10
- (None) #11
- (None) #12
- (None) #13
- (None) #14
- (None) #15
- (None) #16
- (None) Other
- Configure Aggregates

**Total Synthesis**

	Throughput	Reception	Received Pkts	Matching Pkts	Sent Pkts	Throughput	Transmission
From A to B	0.00 b/s	0 p/s	0 p	0 p	0 p	0.00 b/s	0 p/s
From B to A	0.00 b/s	0 p/s	0 p	0 p	0 p	0.00 b/s	0 p/s

Alarms: CPU Usage 3%

### Statistics & Alarms

Different statistics are calculated and displayed by **NetDisturb**:

- for each IP Flow (and for both directions)
- Statistics synthesis by Flow
- Total synthesis & Alarms

These statistics can be saved in a file for a later use.

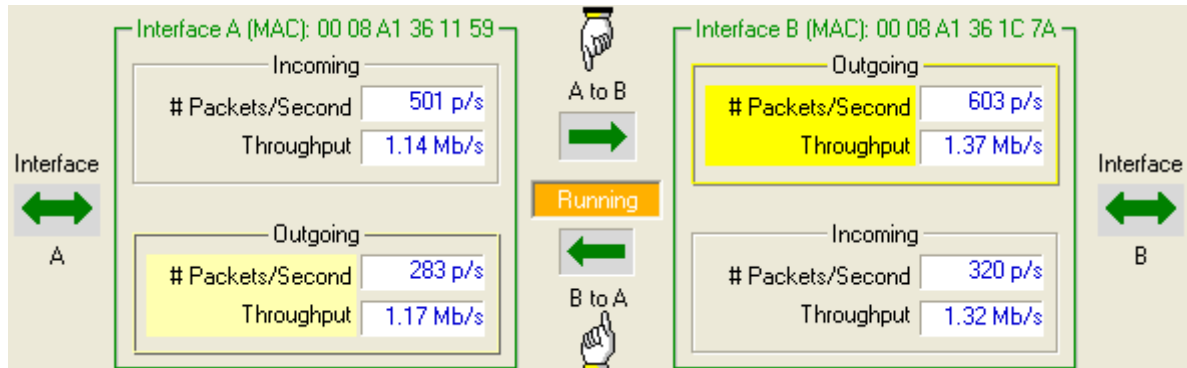
### Statistics for each IP Flow

For each direction (**A → B** or **B → A**) **NetDisturb** displays:

- number of packets matching the mask
- number and percentage of lost or duplicated packets
- number and percentage of delayed packets

Mask [+ Trigger]	Loss & Duplication Law	Delay & Jitter Law
UDP 2009	Lost (1/5) then Duplicate (1/3)	Delay (5) + Jitter (25)
Mask <input type="button" value="Edit"/>	Loss then Duplicate <input type="button" value="Edit"/>	Constant Delay & Exponential Jitter <input type="button" value="Edit"/>
# Incoming Packets 39769	# Lost or Duplicated Packets 23981 [60 %]	# Delayed Packets 31816 [80 %]

and a complete view of traffic statistics (number of packets and throughput) over the **A** and **B** interfaces as shown below:



## Statistics Synthesis by Flow

The synthesis for all IP Flows displays for each flow and for each direction:

- the incoming throughput and number of received packets per second
- the number of packets matching the mask
- the number of lost packets
- the number of delayed packets
- the outgoing throughput and number of sent packets per second

**NetDisturb Client - Impairment Tool for IP Networks - Version 4.4.WSX**

File Edit Actions Working Modes Statistics Help Show Aggregates >>

**IP Flows**

IP Flows	%	Incoming Throughput	Incoming Pkts	Lost Pkts	Delayed Pkts	Outgoing Pkts	Outgoing Throughput
#01 A to B	29	1.14 Mb/s	501 p/s	47843	5742 [12 %]	42101 [88 %]	42099
#01 B to A	10	703 Kb/s	166 p/s	16059	0 [0.0 %]	16059	703 Kb/s
#02 A to B	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#02 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#03 A to B	27	1.10 Mb/s	483 p/s	45954	2297 [5.0 %]	43657 [95 %]	43656
#03 B to A	10	1.05 Mb/s	175 p/s	15594	0 [0.0 %]	15594	1.05 Mb/s
#04 A to B	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#04 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#05 A to B	3	518 Kb/s	45 p/s	4241	0 [0.0 %]	4240	518 Kb/s
#05 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#06 A to B	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#06 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#07 A to B	2	533 Kb/s	46 p/s	4187	208 [5.0 %]	3979 [95 %]	3979
#07 B to A	10	1.11 Mb/s	175 p/s	16492	1978 [12 %]	14514	0.98 Mb/s
#08 A to B	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#08 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#09 A to B	2	534 Kb/s	46 p/s	4178	2529 [61 %]	3343 [80 %]	5037
#09 B to A	10	1.06 Mb/s	167 p/s	16147	0 [0.0 %]	16147 [100 %]	1.06 Mb/s
#10 A to B	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#10 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#11 A to B	2	528 Kb/s	45 p/s	4130	41 [1.0 %]	4089	527 Kb/s
#11 B to A	10	1.09 Mb/s	172 p/s	16154	161 [1.0 %]	15993 [99 %]	1.07 Mb/s
#12 A to B	29	0.00 b/s	0 p/s	48841	29441 [60 %]	39073 [80 %]	58746
#12 B to A	19	0.00 b/s	0 p/s	31178	3742 [12 %]	27436	0.00 b/s
#13 A to B	2	516 Kb/s	44 p/s	4078	485 [12 %]	3593 [88 %]	3593
#13 B to A	10	1.11 Mb/s	171 p/s	16017	0 [0.0 %]	16017 [100 %]	1.10 Mb/s
#14 A to B	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#14 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#15 A to B	2	522 Kb/s	45 p/s	4060	41 [1.0 %]	4019	516 Kb/s
#15 B to A	10	1.15 Mb/s	178 p/s	15961	0 [0.0 %]	15961 [100 %]	1.14 Mb/s
#16 A to B	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
#16 B to A	0	0.00 b/s	0 p/s	0	0 [0.0 %]	0	0.00 b/s
----- A to B	0	1.03 Kb/s	0 p/s	26	0 [0.0 %]	0	26
----- B to A	10	1.10 Mb/s	174 p/s	16612	0 [0.0 %]	16612	1.10 Mb/s

**Total Synthesis**

	Throughput Reception	Received Pkts	Matching Pkts	Sent Pkts	Throughput Transmission
From A to B	5.31 Mb/s	1252 p/s	168223 p	170169 p	5.17 Mb/s
From B to A	8.34 Mb/s	1373 p/s	160435 p	160425 p	8.17 Mb/s

**Alarms**

CPU Usage: 96 %

Statistics Synthesis by Flow - example

## Total synthesis

At the bottom of the Client window, the total synthesis displays the following parameters for both directions (A → B or B → A):

- Throughput and number of packets per second received
- Number of packets received
- Number of matching packets
- Number of packets sent
- Throughput and number of packets per second transmitted

Total Synthesis							Alarms
	Throughput Reception	Received Pkts	Matching Pkts	Sent Pkts	Throughput Transmission		
From A to B	1.16 Mb/s	491 p/s	28675 p	28675 p	28381 p	1.12 Mb/s	487 p/s
From B to A	4.16 Mb/s	745 p/s	81630 p	81504 p	81630 p	4.16 Mb/s	745 p/s

CPU Usage: 18 %

## Alarms

The alarms encountered by the **NetDisturb** driver can be displayed by the user and are classified per direction for both interfaces:

<i>Incoming direction</i>	<i>Outgoing direction</i>
<ul style="list-style-type: none"> <li>• Number of lost packets</li> <li>• Number of lost bytes</li> <li>• Number of errors returned by the Driver at the Interface</li> <li>• Number of missing buffers to keep packets</li> <li>• Number of ignored flows (when the multi-flows option is active).</li> </ul>	<ul style="list-style-type: none"> <li>• Number of lost packets</li> <li>• Number of lost bytes</li> <li>• Number of errors returned by the Driver at the interface</li> </ul>

**NetDisturb Client - Alarms Summary**

Alarms Linked to the Direction from Interface A to Interface B

Incoming from A

- # Packets Lost: 0
- # Bytes Lost: 0
- # Driver Errors: 0
- # Buffer Missing Errors: 0
- # TCP/UDP Connections Lost: 0

A to B

Outgoing to B

- # Packets Lost: 0
- # Bytes Lost: 0
- # Driver Errors: 0

Details

Alarms Linked to the Direction from Interface B to Interface A

Outgoing to A

- # Packets Lost: 0
- # Bytes Lost: 0
- # Driver Errors: 0

B to A

Incoming from B

- # Packets Lost: 0
- # Bytes Lost: 0
- # Driver Errors: 0
- # Buffer Missing Errors: 0
- # TCP/UDP Connections Lost: 0

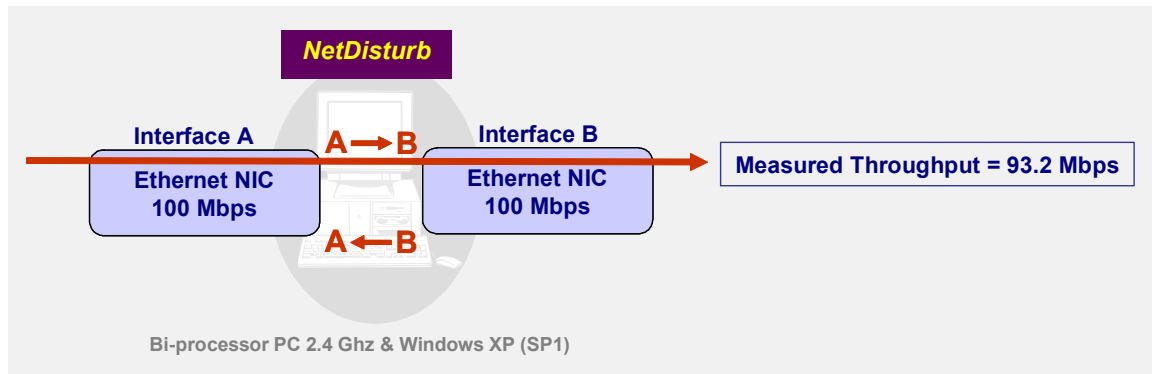
Details

OK Clear Alarms Update Alarms Summary

## Performances

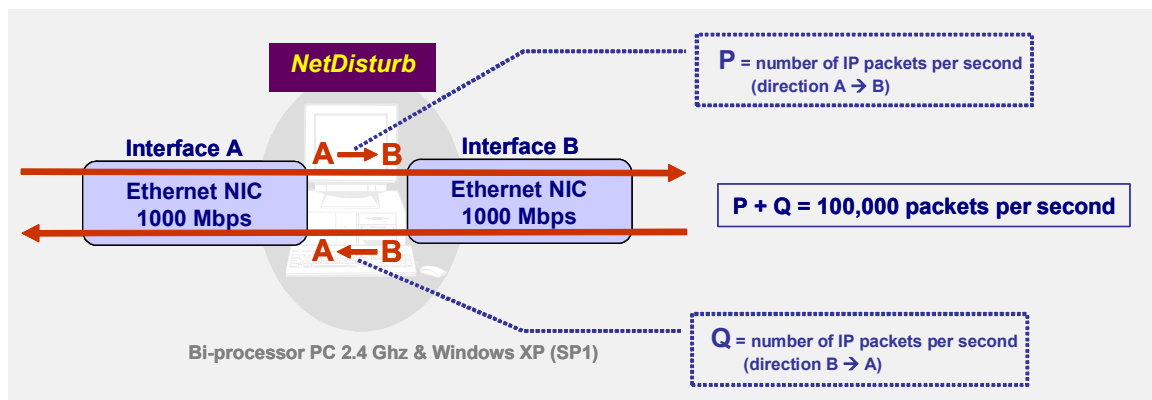
To illustrate the key performances of **NetDisturb**, 2 examples are presented hereafter (by using a bi-processor PC 2.4 Ghz with windows XP SP1).

### Example 1: use of 2 Fast Ethernet NICs



**NetDisturb** is configured with 16 IP flows (no loss and no delay for each flow). With Fast Ethernet NICs, the throughput measured is 93.2 Mbps in one direction.

### Example 2: use of 2 Gigabit Ethernet NICs



By using 2 Gigabit NICs, **NetDisturb** can handle up to 100,000 packets per second with 16 IP flows defined (for both directions).

These two examples show some performances of **NetDisturb**. This will avoid heavy investments in expensive hardware solutions.

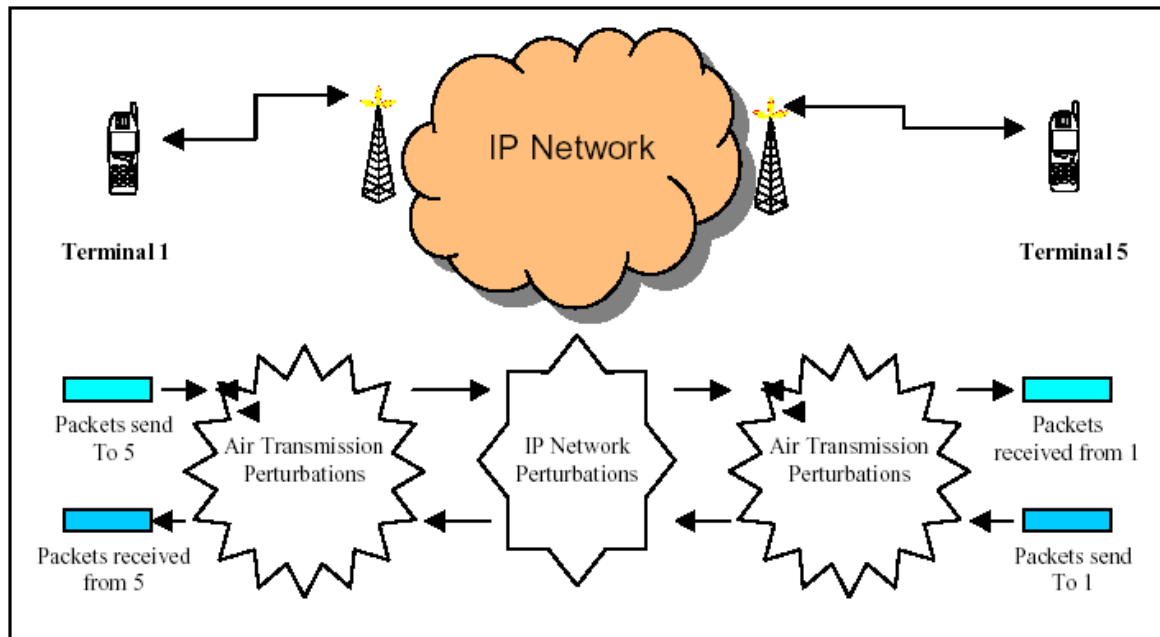
### Applications

- *Performance & Acceptance Tests:* Qualify and evaluate the behavior of IP equipments (phone, fax, gateway, etc.) and applications (audio and video streaming, etc.) on IP networks.
- *Configuration and control of IP Equipments for product verification and test:* Define different QoS levels in an Intranet or Internet environment to configure terminals, gateways and routers.
- *Test Laboratories:* **NetDisturb** provides repeatable QoS on different flows using configuration mode and values (loss, duplicate, delay) defined by the user, and so re-create real world problems in the lab.
- *Applications test:* **NetDisturb** allows testing applications such as Voice over IP, streaming audio and video, and other distributed applications.
- *Emulation of symmetric or asymmetric network conditions (Lan, Man, Wan):* latency, jitter, packet loss, bandwidth limitations, etc. to test IP applications (VoIP, streaming audio & video, etc.), services and products sensitive to various real conditions.

#### *Some publications mentioning the use of **NetDisturb***

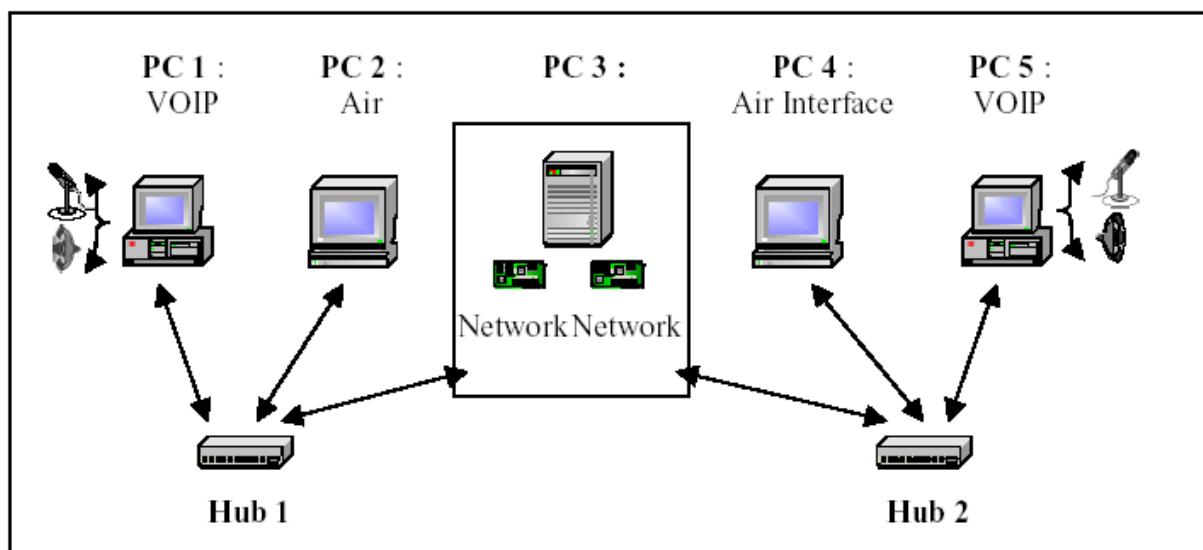
- The Communications and Information network Association of Japan (CIAJ) which represents manufacturers supplying network devices and terminals has published a report on 2002: Report on speech quality investigation of VoIP Terminals (gateways and IP phones):  
[http://www.ciaj.or.jp/tusin/pressrelease/voip\\_1e.html](http://www.ciaj.or.jp/tusin/pressrelease/voip_1e.html)  
"We adopted **NetDisturb**, ... as a network simulator because of its ease of installation and operation in Windows".
- 3GPP Technical Specification Group Services and System Aspects TSG-S4
  - Test Plan for the Adaptative Multi-Rate Wide-Band (AMR-WB) and Narrow-Band (AMR-NB) in packet switched networks.
  - Test Plan for 3G packet switched conversation tests (comparison of quality offered by different speech coders over packet switched networks)  
**NetDisturb** is used as the simulated network.

The following illustrations describe the system that is simulated for these tests.



*Packet switch audio communication simulator*

This is simulated by using 5 PCs as shown below, with PC# 3 using the **NetDisturb** software as network simulator.



*Simulation platform*



### Customer references

Present on the market since 1998, **NetDisturb** is used in more than 40 countries.

See some worldwide references of satisfied customers:

Alcatel, ANZ Bank, AT&T, Bell Canada, Cisco, Commtech Wireless, Department of Defense, Equant, France Telecom, Gensight, Global Crossing, Iwatsu, Juniper, Motorola, Nortel Networks, NEC, NTT, Panasonic, Philips, PIKA Technologies, Polycom, Psytechnics, Raytheon, Schlumberger, Scopus, Tekelec, TF1, Toshiba, UTStarcom, WL Gore, Xerox, etc. as well as many universities and telecom institutes.

### Conditions of use

**NetDisturb** is licensed on a per workstation basis. You will need to purchase a separate license for each machine that you install it on.

Each licensed copy of the software installed on a workstation has a unique Site Code that requires the corresponding unique Site Key to be entered before being operational.

### Delivery

Includes CD with documentation, printed installation guide, technical support and software maintenance (including major and minor software upgrades) for a period of twelve months from the date of purchase.

<p>To download an evaluation of <b>NetDisturb</b> please visit us at: <a href="http://www.zti-telecom.com/pages/main-ip.htm">http://www.zti-telecom.com/pages/main-ip.htm</a></p>
---

## Glossary of Terms

### Bandwidth Throttling

Bandwidth throttling is used for two main purposes:

- Quantify network resources - by evaluating the application's bandwidth requirements, network managers can determine in advance the amount of bandwidth to purchase.
- Evaluate QoS mechanisms - prior to a decision on which QoS mechanism is appropriate for the enterprise, network managers can emulate different Service Level Agreements and evaluate the ROI of different services such as Frame Relay, Diffserv etc.

### Delay jitter

Delay variation of the packet transfer caused by the queuing and access delays in the source node, all transit node delays, and the receive buffer delay in the destination node.

### IP Flow

A flow is a set of packets with a set of common packet properties. The IP flow can be uni-directional or bi directional and is defined by n-tuple (typical case: 5-tuple – IP source address, IP destination address, Source port number, destination port number, and transport type).

### Jitter

In data networks, jitter refers to packet jitter, not bit jitter and represents the variation in a stream's delay (expressed in seconds). Jitter is the standard deviation of delay and is one of the IP performance metrics.

The jitter is the absolute value of the difference between the delay measurement of two packets belonging to the same stream. The jitter between two consecutive packets in a stream is reported as the "instantaneous jitter". Instantaneous jitter can be expressed as  $|D(i+1) - D(i)|$  where D equals the delay and I is the test sequence number. Packets lost are not counted in the jitter measurement.

Jitter particularly affects the performance of real time network applications such as streaming video and audio. In these types of applications, data needs to arrive at a specific time frame or it becomes useless. As a result, many streaming audio and video application can be severely impacted by high jitter.

### Latency (End-to-End Delay)

Latency is defined as the period of time it takes for the information element (voice, e-mail, web, etc.) to traverse the network from its origin to its destination. For basic data where a small delay can be tolerated, latency is usually not an issue. However, for communications services used for videoconferencing or VoIP for example, latency can interfere with the audio and/or visual communications. In shared bandwidth transmission environments, it is possible to encounter latency which varies dynamically, caused by perhaps a single user accessing or originating multi-megabyte-sized files or accessing high bandwidth streaming signals.

When discussing network latencies relative to the operation of H.323, there are 3 general categories to consider:

- End-to-End latency in a given direction. This category addresses the total transit time for data of a given data stream to arrive at the remote endpoint.
- Intra-stream latency. This category addresses latencies within a given data stream which boils down to inter-packet latencies that deviate outside of the normal transmit time by more than a predefined value.
- Inter-stream latency. This category addresses the relative latencies that can be encountered between the audio and video data streams.

### Network Errors

Generally, packet losses or corruptions are the source of the network errors:

- Main cases of packet loss:
  - Network load - which can cause a packet queue in a network hop to overflow. This will cause new packets to be dropped due to lack of memory space. This typically results in a burst loss where several packets from one endpoint are lost at once.
  - Limited bandwidth - QoS parameters such as Frame Relay CIR (Committed Information Rate) or Diffserv bandwidth polices can define a data rate limit which, when exceeded, can result in dropped packets.
  - Congestion avoidance mechanisms, such as RED (Random Early Detection) implemented in network gateways and routers can selectively decode and drop packets in order to avoid what seems to be an upcoming congestion trend.
  - IP header corruption is an error that creates a malformed IP header. A malformed IP header will cause the next router receiving the corrupted packet to drop it.
  - Hardware faults such as link disconnections and device shutdown.
- Packet corruption: is caused by errors in the physical layer, which in turn causes data bits to toggle.

### Network Impairment

Network impairment is the process of interfering with network traffic for the purpose of testing and evaluating the overall performance of TCP/IP networks. Due to TCP/IP's dynamic routing algorithms, packets may be delayed, reordered, duplicated, fragmented or even lost.

### Out Of Sequence Packets (OOS)

Out of sequence packets typically occur when the packet stream is transmitted over multiple paths of unequal delay to a particular endpoint. Packets may arrive at the destination with incorrect ordering.

### Packet Loss

Packet loss is a normal phenomenon on packet networks: when data transmitted from an originating device don't arrive at the intended destination. Loss can be caused by many different reasons: overloaded links, excessive collisions on a LAN, and physical media errors, to name a few. Transport layers such as TCP account for loss and allow packet recovery under reasonable loss conditions.

### Propagation Delay

The propagation delay is the time required for a packet to travel over the network (difference between the transmission of data to its receipt at the other end).

### Quality of Service (QoS)

A list of measurable attributes that should be met for a specific communications service on a network: bandwidth, latency, packet loss rate, packet desequencing and latency variation (jitter) for real-time applications such as VoIP, and service availability.