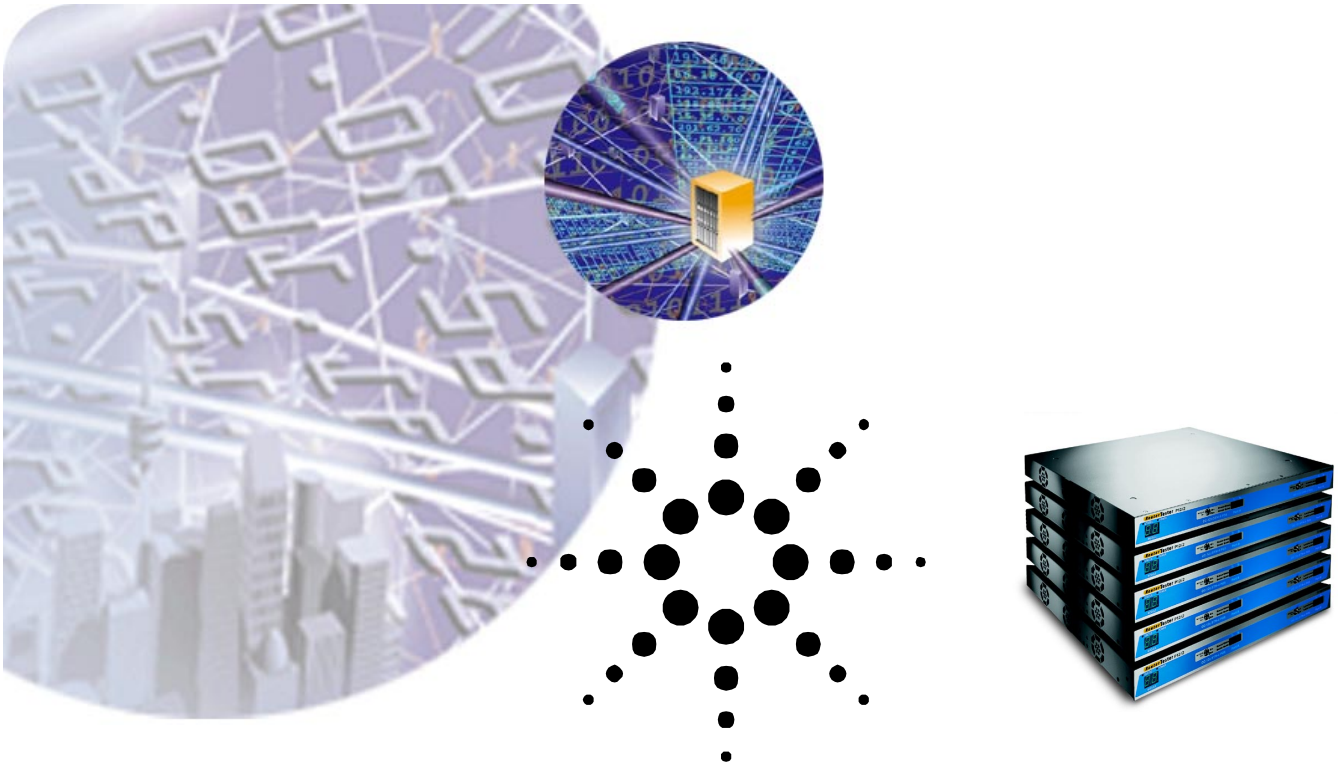


Understanding IP Analysis

Agilent Technologies RouterTester Application Note

Introduction

The IP Analysis Test Software adds powerful data capture and analysis capabilities to the Agilent Technologies RouterTester. These new offline tools complement the real time statistics already available. The need for such tools is clear: real time statistics will expose performance and functional issues of a system under test (SUT), but offline capture and analysis is needed to investigate issues at the fine resolution that simply cannot be provided in real-time.

Most customers are familiar with the basic concepts of capture and decode presentation of the capture data. Many existing test tools, such as the Agilent Broadband Series Test System (BSTS), include these capabilities, together with advanced features such as pattern match filtering and triggered capture.

The IP Analysis application delivers all the features you would expect from an industry leading test tool. But as capture memory sizes and the number of test ports have increased it has become apparent that the old use paradigms no longer work. There is simply too much data for a standard decode viewer to handle. Consider a fully

configured RouterTester with 64 ports, each supporting 64 MB of capture memory. With such a configuration it is easy to capture around 4.0 GB of data! How could a user ever find the needle of interest in such a huge haystack of data?

A design goal of the IP Analysis application is to provide the user with a graphical picture of an entire capture buffer. The user selects a point of interest, such as a spike in latency, and the system "drills down" by re-analyzing the location around the selected point at a much finer resolution. The user can continue to scale in at finer resolutions until he or she has located a single packet of interest out of the possible two million in the capture buffer. It is this ability to "find the needle in the haystack" that makes the IP Analysis application unique in the industry.

At a Glance

There are two major parts to the IP Analysis application:

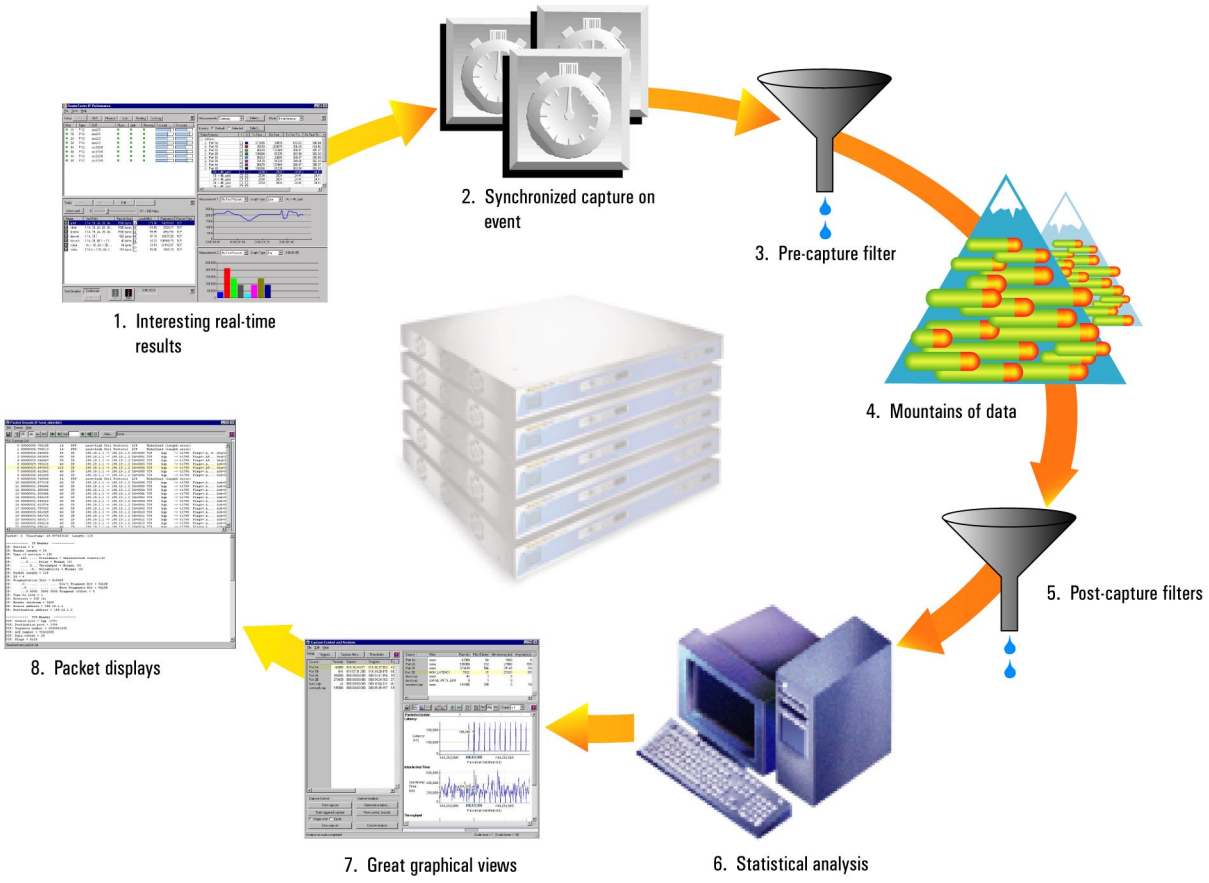
- capture setup and control
- capture results analysis

Correspondingly, the application main window is divided down the middle. The left hand side is dedicated to capture setup and control. The right side is dedicated to capture analysis presentation.

The application guides the user through the following task flow.

Configuration and setup

Traffic flows are configured and set up using the existing IP Performance application. The IP Analysis application is concerned only with received data streams. During an IP Performance test, the user might notice real time statistics that merit further investigation.



Capture

Capture can be started either manually or by a start trigger. Capture may also be stopped manually or by a stop trigger. Triggers are events detected by the hardware. There are a range of triggers available, including:

- pattern match
- layer 2 and layer 3 errors, such as TCP checksum error
- thresholds. Threshold triggers fire when a measurement, such as latency, falls outside a pre-defined threshold range.

Capture can also be configured to stop when full, or to continue capturing indefinitely by overwriting the oldest data first. If stopped by a trigger, capture will continue to capture until the trigger point is centered in the buffer. This is to provide capture data both before and after the trigger event.

Precaptured filtering

The IP Analysis application includes a range of precapture filtering tools to allow the user to store only packets of interest. Packets may be selected for store or discard based on up to six pattern matchers, and/or a range of status filters that can select packets of interest flagged by the hardware. In particular, the user can request that the application only store those packets that match the streams selected for real time viewing of statistics.

Synchronized capture

Even with precapture filters in place, a great deal of data can be generated. This becomes clear when you consider that the RouterTester is a multi-port system and that a synchronized capture may be performed on up to 128 ports at a time. It is at this point that we turn to the analysis side of the application to help make sense of the data.

Post-capture filter

The application provides a powerful post-capture filter, with a super-set of the capabilities of the pre-capture filter. A post-capture filter may be constructed from any combination of patterns, value threshold filters and status filters. The need for a post-capture filter becomes clear when you consider that a full understanding of the captured data may require a number of different post-capture filters to be applied successively to the same data.

For example, a user may want to investigate the effect of different DS byte values on packet latency. He/she does this by successively analyzing the data with a different post-capture filter, each selecting a different DS value.

Post-capture statistics

A sophisticated analysis engine runs on the embedded microprocessor dedicated to each port. This software generates a range of post-capture statistics and can process approximately 50,000 packets per second. The user can generate measurements for one or more selected ports and in a few seconds the results are ready. Two basic types of analysis are available: value by time and distribution by value.

An instance of the analysis engine is also provided on the host PC so that users may analyze saved capture files in exactly the same way as live data.

Graphical analysis

Key statistics such as latency, interarrival time and packet length are presented as line graphs. Up to four traces may be plotted simultaneously on each graph for comparison between ports, regression analysis or to study the effect of post-capture filter settings.

Distribution charts are also available for these three statistics. The user may scale in for a more detailed analysis at any selected point on a graph or expand a selected bar of a distribution chart.

Protocol decode

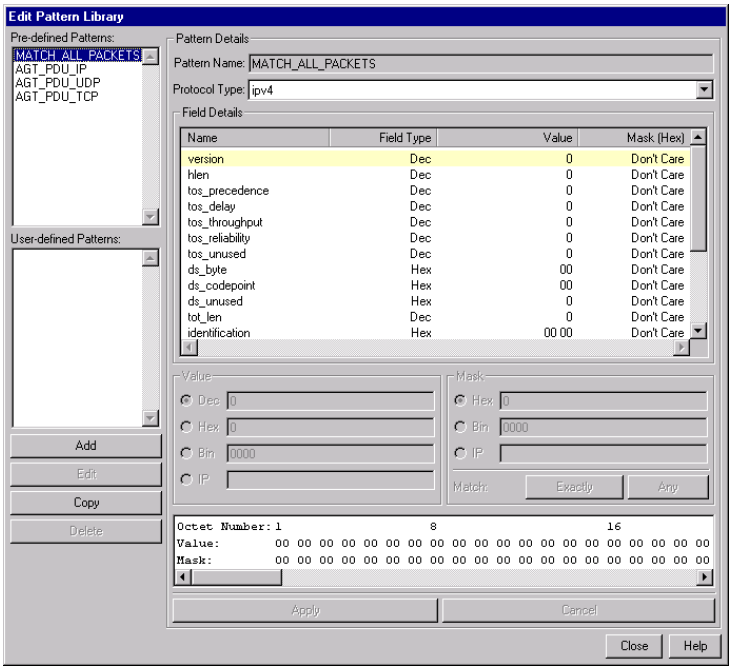
The packet display provides a comprehensive protocol decode view of the capture data. A wide range of TCP/IP protocols are supported. The packet display uses the same decode engine used by the Agilent Internet Advisor and BSTS, and decodes for the same large range of IP protocols are available. The user may proceed to the packet display directly from the capture source, or use the analysis results to help navigate to the packet of interest.

The Pattern Library

Patterns are used extensively by the IP Analysis application, for configuring pre-capture filters, capture triggers and post capture filters. A pattern is simply a 128 byte array and an associated mask of the same length.

Patterns are matched against up to the first 128 octets of each IP packet. Only those bits selected by the mask need match the pattern, all other bits are ignored. This facility gives plenty of scope for matching any fixed field in the header portion of each packet.

The IP Analysis application provides a pattern editor for creating and modifying patterns, and a pattern library for storing previously created patterns.



The pattern editor allows the user to manipulate the pattern bits directly, but mostly the user will create and modify patterns by selecting a protocol and entering a desired value for a field. For example, the user would enter '20' into the source_port field of the TCP protocol to create a pattern that selects packets sourced from port 20.

To keep the pattern editor simple, but at the same time give maximum flexibility to the user:

- The user selects the protocols to use on each pattern. A pattern does not have a pre-assigned protocol.
- Each protocol is edited independently. For example, the user may select IP to edit IP fields, then select TCP to edit TCP fields on the same pattern. Each protocol has a default offset from the start of the packet, which may be overridden by the user.
- Only fixed length field types are supported

Protocols implemented currently are IP, TCP, UDP and some BGP-4 fields. However, all protocol knowledge is loaded in by the application from an external text file in XML format. The

customer or support engineer is at liberty to add their own protocol implementations to the protocols.xml file, which can be found at c:\Program Files\Agilent\RouterTester\protocols.

Protocols added to the protocols.xml file may be used to edit patterns in the GUI and the System API in exactly the same way as the supplied protocols.

Analyzing Captured Data

There are many features worth discussing relating to obtaining a capture, such as the various pre-capture filters and triggers. However, for the sake of brevity, I will concentrate in this document on the new concepts behind capture analysis.

Analysis data is generated by selecting a source port or saved capture file and pressing “Generate Measurements”. The resulting data is stored in an entity known as an Analysis Set. Each analysis set is represented by a line entry in the table at the top right of the IP Analysis application. An analysis set will remain until the user deletes it. Multiple analysis sets may be generated from the same source data. The usual reason for doing this would be to apply different post-capture filters.

As mentioned before, the large potential volumes of capture data cannot be dealt with by a protocol decode viewer - a graphical picture of the data is necessary to help locate points of interest for further investigation. Once an interesting spike or dip on a graph is found, the user must be able to dig deeper and investigate what is going on at an ever finer resolution.

Consider a full capture buffer of mostly minimum length (40 byte) IP packets. Such a buffer may contain, say 1,000,000 packets. For each packet, we may be interested in a number of statistics, such as:

- Latency: the amount of time between the transmit time from a RouterTester port to the receive time at a RouterTester port.
- Inter -Arrival time: the interval of time between each packet received
- Packet Length

If we made these three measurements for each packet, we would have 3,000,000 data points for the capture buffer.

Consider now, that we may want to make the same measurements for up to 64 capture buffers in a RouterTester system. Obviously, the volumes of data would be huge - and attempts to transmit and collate these large amounts of data could well overload the RouterTester.

The purpose of collecting this data is to plot it on a graph. And since we can only fit about 500 or so points along the time axis of the graph, some form of data reduction is necessary. One possible approach might be to “sample” only a limited number of packets, but this could easily lead to the “interesting” data being missed. Instead, the IP Analysis application analyses every packet, and reports minimum, maximum and average values for clusters of packets.

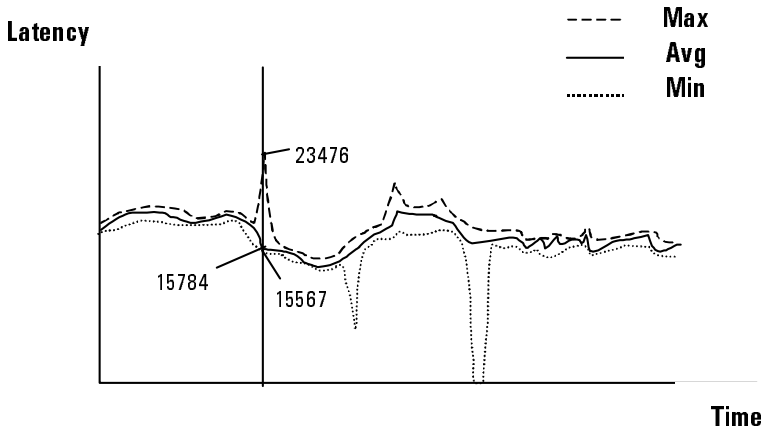
This is how it works: The graphical client can only plot 500 or so points, so that's what it asks for: 500 data points, or clusters. The analysis engine responds by dividing the number of packets in the source by the requested number of clusters to determine how many packets comprise each cluster. The analysis

engine then analyses every packet, and computes the minimum, maximum and average values for each statistic over each cluster.

On our graph representing 1,000,000 packets, only 500 points are actually plotted and each point represents a cluster of 2,000 packets. You will notice that there are three buttons labelled: Min Avg and Max. By toggling these buttons, you can see the spread of data provided by each cluster. If you have the Max button selected, for example, you can be sure that there is no packet in the capture buffer that exceeds the values plotted on the graph at each point.

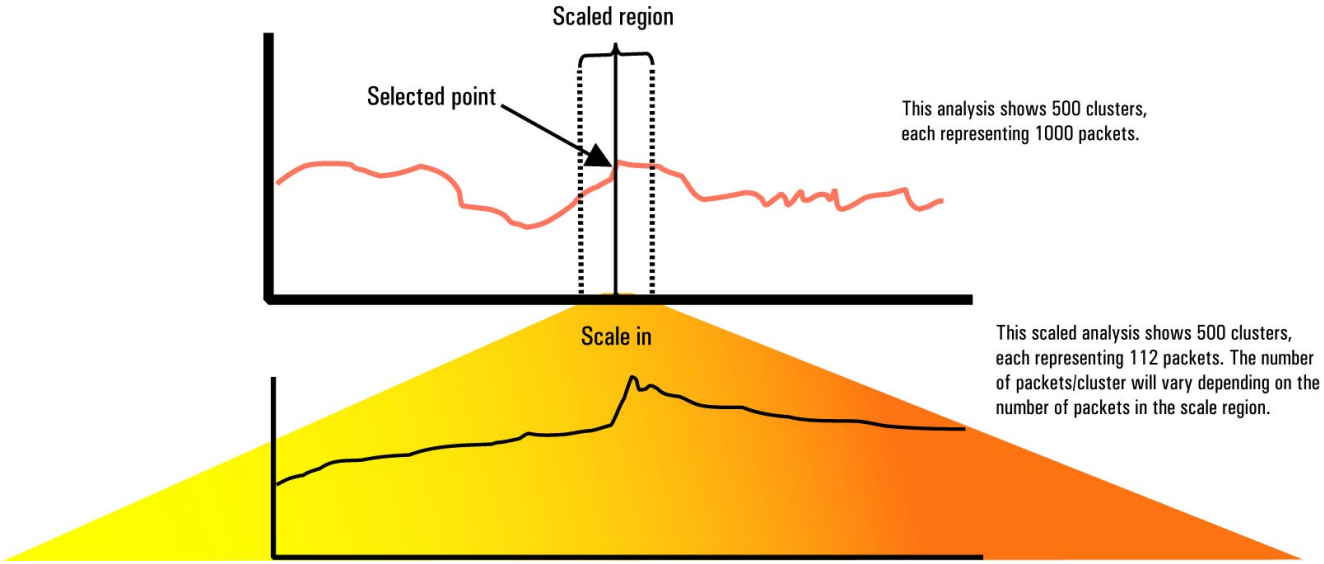
Packet and Cluster Statistics

The IP Analysis application computes the three statistics described above for each packet in the cluster. These statistics are known as packet statistics. In addition, a number of statistics are computed for the cluster as a whole. These statistics are known as cluster statistics. A key difference between a cluster statistic and a packet statistic is that cluster statistics report only one value per cluster, where as packet statistics report three: minimum, average and maximum.



Each point on a graph of latency vs. time is a cluster. A cluster represents one or more packets — potentially thousands of packets. The minimum and maximum traces plot the extreme values for each cluster. The user can select any cluster to see the values at the selected point.

Packet statistics	• Latency (ns) • Interarrival Time (ns) • Packet Length (octets)
Cluster statistics	• Bandwidth (octets / second) • Throughput (packets / second) • Packet Loss (packets /cluster) • TCP packet count (packets /cluster) • UDP packet count (packets /cluster) • Checksum error count (packets /cluster) • Misdirected packet count (packets /cluster) • Unexpected Stream ID packet count (packets /cluster) • Sequence Error count (packets /cluster) • IP Header error count (packets /cluster) • Fragmented IP header count (packets /cluster) • No “Test Payload” count (packets /cluster)



Scaling

In the previous discussion, we discovered that the application plots values for clusters which may each represent many, even thousands of, packets. In order to view these plots clearly, the user may select from a range of graphical zoom factors. However, graphical zooming does not expose data at a finer resolution.

The user may notice that at one location on the graph, latency is particularly high and therefore wish to "drill down" and find out what is happening at a finer resolution. The IP analysis application provides for this through the scaling feature. To increase scale, the user selects a point on the graph and asks the system to increase scale at that point.

The system responds as follows:

- a time interval is computed that is 10% of the interval covered by the original graph and that is centered on the selected point
- a new analysis is generated for packets within the selected interval.

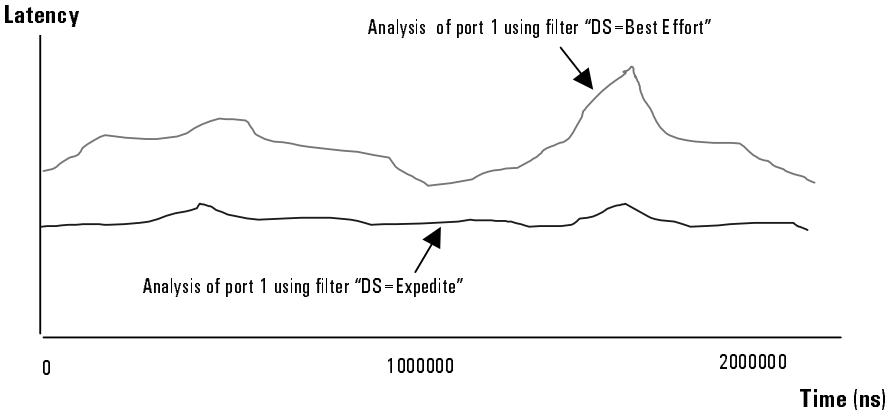
The important thing to note is that the number of packets per cluster will decrease each time the user scales in. For example, the original graph might plot 500 clusters each representing 1,000 packets. After scaling, we might see 500 clusters each representing only 100 packets. The actual number of packets per cluster will depend on the number of packets encountered in the scale region. If we scale far enough, we will achieve one packet per cluster, i.e. each point represents one packet. In this case, the Min, Max and Avg traces will all be identical.

Once a suitably fine resolution has been achieved, the user is able to switch to a protocol decode presentation of the packet data at a selected point.

The diagram above illustrates these concepts further. The user may increase and decrease scale at different selected points as much as is needed to gain the required understanding of the capture data.

The Post Capture Filter

When an analysis is requested, the user is given an opportunity to specify a post capture filter. You might at first wonder why a post capture filter is needed when a powerful pre-capture filter is also available. The major reason is that different post capture filters can be used successively on the same source data. Consider, for example, a requirement to investigate the effect of different DiffServ DS byte values on Latency. One way to achieve this is to generate several analysis sets, each using a different post-capture filter that filters IP packets with a different DS byte value.



Overlaying Plots

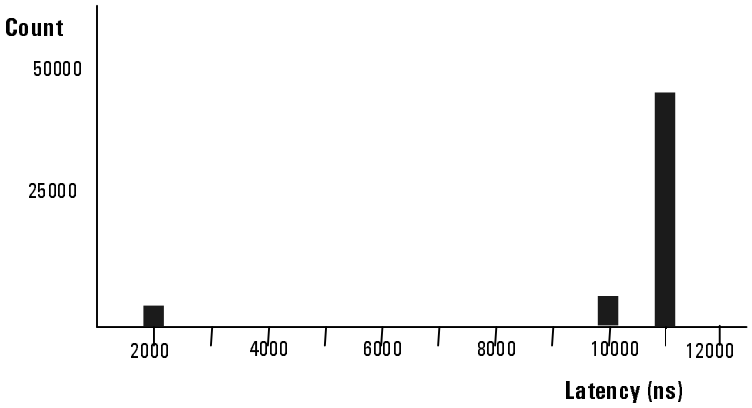
A very important analysis feature is the ability to overlay up to four traces on the one set of graphs. In the previous example, the user will want to plot the Latency traces for the different DS byte values on the same chart.

Another reason for overlaying plots is for regression testing. A customer might want to compare current results with those obtained with the prior version of his hardware.

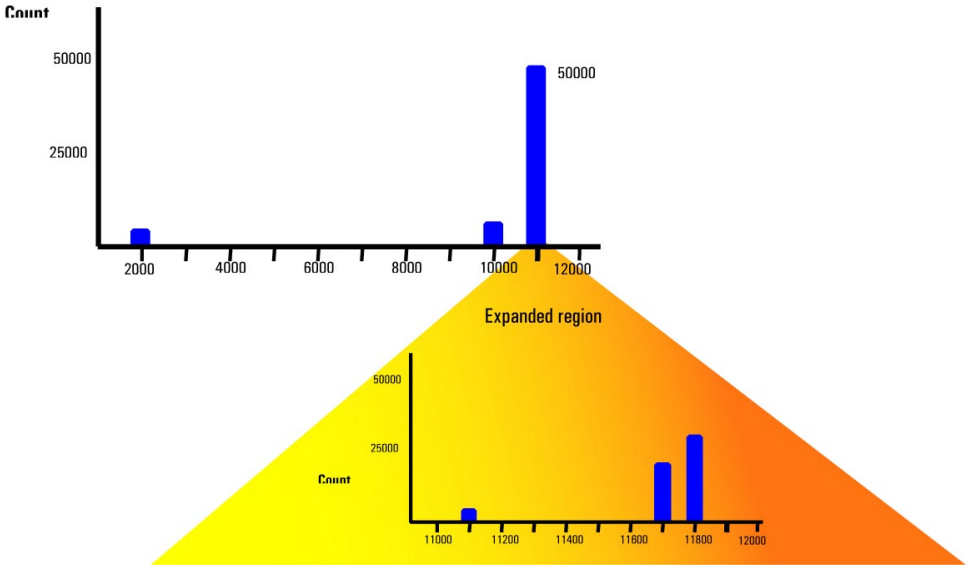
A question that needs to be resolved when overlaying plots is how to align the different traces. This task is simple if the traces are sourced from the same synchronized capture, but what if the user wants to compare data from a capture just taken with a reference capture file that was taken three months ago?

To allow data from disparate sources to be plotted together, the time axis is adjusted dynamically. Two important things to note are:

- graphs are aligned to the right by capture stop time
- the time scale is adjusted so that the first cluster from the longest capture is marked as time = 0.



Note that it is possible to overlay capture data taken over greatly disparate time scales. For example, one capture may span 0.00001 seconds, while another may have taken several hours. Clearly, if the analysis results for these two captures were plotted together, the short capture would simply appear as a single vertical line at the right hand side of the graph. The application allows any four sources to be overlaid in a plot — it is left to the user's judgement to determine when it makes sense to do so.



Distribution Analysis

The IP Analysis Application provides two different views of these statistics:

- line graphs showing the value over time
- and distribution charts showing how the range of values is distributed.

A distribution chart is illustrated above.

The calculation of distribution data requires two inputs, the number of distribution buckets and the range of values allocated to each bucket. The IP Analysis application by default will calculate ten buckets, and the range of values is extracted from the analysis set data. For this reason, it is necessary to generate measurements for the cluster graphs first, before generating a distribution. If multiple analysis sets are selected when generating a distribution, the range of values will encompass those for all the selected analysis sets.

Consider an example. The minimum latency encountered by analysis set 1 is 2,000 and the maximum is 12,000. A distribution chart generated for this analysis set will be over a latency range of 2,000 to 12,000. If the user wants a combined distribution with analysis set 2, which has a minimum latency of 5,000 and a maximum of 17,000, the combined distribution will be over a latency range of 2,000 to 17,000.

On the chart itself, it is the minimum value for each bucket that is shown on the horizontal axis.

Distribution Expansion

It is unusual for real data to present an even distribution. Normally, most packets will be represented by just one bar. To glean real meaning, it is likely that the user will want to "expand" the distribution to see how the packets are distributed within the range represented by a selected bar.

Firstly, the user can select any bar to see the value for that bar. Secondly, the user can request the selected bar be expanded to generate a new distribution over the value range of the selected bar. As with scaling, the user may repeatedly expand and contract selected bars as often as needed to attain a full understanding of the underlying capture data.

Summary

The IP Analysis application goes far beyond the capture/decode paradigm seen on other test equipment. The competitive differentiators are many, including:

- multi-port synchronized capture of up to 128 ports
- powerful pre- and post-capture filtering
- threshold triggering to start and stop capture on a performance event
- distributed analysis engine runs simultaneously on up to 128 ports. Analysis also available for saved capture data.
- line graphs and distribution charts for Latency, Interarrival Time and Packet Length.
- multiple analyses on the same source data. Overlay the plots and compare!
- regression analysis possible by analyzing a reference saved capture and overlaying with current data
- drill down to see analysis and distribution data at whatever resolution is required to find the information needed. A full solution for navigating to a point of interest and pin-pointing a problem.
- A comprehensive decode viewer for TCP/IP protocols - navigate by selecting a point on a graph

Acronyms

API	Application Programming Interface
BGP-4	Border Gateway Protocol, Version 4
BSTS	Agilent Technologies Broadband Series Test System
DS	Differentiated Services
DiffServ	Differentiated Services
GUI	Graphical User Interface
IP	Internet Protocol
SUT	System Under Test
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
XML	Extensible Markup Language

Agilent Technologies RouterTester

RouterTester provides true Internet-scale testing through realistic routing protocol support, multi-stream wire-speed traffic generation and real-time analysis, and multi-port scalability. RouterTester is set to grow as the testing needs of the carrier class router industry evolve to meet the challenges of scale and Quality of Service within the Internet.

www.Agilent.com/comms/RouterTester

United States:
Agilent Technologies
Test and Measurement Call Center
P.O. Box 4026
Englewood, CO 80155-4026
1-800-452-4844

Canada:
Agilent Technologies Canada Inc.
5150 Spectrum Way
Mississauga, Ontario
L4W 5G1
1-877-894-4414

Europe:
Agilent Technologies
European Marketing Organisation
P.O. Box 999
1180 AZ Amstelveen
The Netherlands
(31 20) 547-9999

Japan:
Agilent Technologies Japan Ltd.
Measurement Assistance Center
9-1, Takakura-Cho, Hachioji-Shi,
Tokyo 192-8510, Japan
Tel: (81) 426-56-7832
Fax: (81) 426-56-7840

Latin America:
Agilent Technologies
Latin American Region Headquarters
5200 Blue Lagoon Drive, Suite #950
Miami, Florida 33126
U.S.A.
Tel: (305) 267-4245
Fax: (305) 267-4286

Asia Pacific:
Agilent Technologies
19/F, Cityplaza One, 1111 King's Road,
Taikoo Shing, Hong Kong, SAR
Tel: (852) 2599-7889
Fax: (852) 2506-9233

Australia/New Zealand:
Agilent Technologies Australia Pty Ltd
347 Burwood Highway
Forest Hill, Victoria 3131
Tel: 1-800-629-485 (Australia)
Fax: (61-3) 9272-0749
Tel: 0-800-738-378 (New Zealand)
Fax: (64-4) 802-6881

Specifications subject to change.

5980-1708E 06/00 Rev A

Agilent Technologies
Innovating the HP Way