Spirent Communications White Paper

Jitter

Version 1.00

March 2007

Spirent Communications, Inc.

26750 Agoura Road
Calabasas, CA
91302 USA

Support Contacts

U.S.
E-mail: support@spirentcom.com
Web: http://support.spirentcom.com
Toll Free: 1-800-SPIRENT (1-800-774-7368)
Phone: + 1 818-676-2616
Fax: +1 818-880-9154

Europe
E-mail: support.europe@spirentcom.com
Web: http://support.spirentcom.com
Phone: + 33 (0) 1 61 37 22 70
Fax: +33 (0) 1 61 37 22 51

Copyright

Limited Warranty

Spirent Communications, Inc. ("Spirent") warrants that its Products will conform to the description on the face of order, that it will convey good title thereto, and that the Product will be delivered free from any lawful security interest or other lien or encumbrance.

Spirent further warrants to Customer that hardware which it supplies and the tangible media on which it supplies software will be free from significant defects in materials and workmanship for a period of twelve (12) months, except as otherwise noted, from the date of delivery (the "Warranty Period"), under normal use and conditions.

To the extent the Product is or contains software ("Software"), Spirent also warrants that during the Warranty Period, the Software which it supplies will operate substantially in accordance with the specifications supplied by Spirent for such Software. Spirent does not warrant that the functions contained in the Software will meet a specific requirement or that the operation will be uninterrupted or error free. Spirent shall have no warranty obligations whatsoever with respect to any Software which has been modified in any manner by Customer or any third party. The Warranty Period for Wireless software products shall be defined as ninety (90) days from the date of delivery.

Defective Products and Software under warranty shall be, at Spirent's discretion, repaired or replaced or a credit issued to Customer's account for an amount equal to the price paid for such Product provided that:
(a) such Product is returned to Spirent after first obtaining a return authorization number and shipping instructions, freight prepaid, to Spirent's location in the United States; (b) Customer provides a written explanation of the defect or Software failure claimed by Customer; and (c) the claimed defect actually exists and was not caused by neglect, accident, misuse, improper installation, improper repair, fire, flood, lightning, power surges, earthquake, or alteration. Spirent will ship repaired Products to Customer, freight prepaid, within ten (10) working days after receipt of defective Products. Except as otherwise stated, any claim on account of defective materials or for any other cause whatsoever will conclusively be deemed waived by Customer unless written notice thereof is given to Spirent within the Warranty Period. Spirent reserves the right to change the warranty and service policy set forth above at any time, after reasonable notice and without liability to Customer.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, ALL IMPLIED WARRANTIES, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT AND FITNESS FOR A PARTICULAR.

PURPOSE, ARE HEREBY EXCLUDED, AND THE LIABILITY OF SPIRENT, IF ANY, FOR DAMAGE RELATING TO ANY ALLEGEDLY DEFECTIVE PRODUCT SHALL BE LIMITED TO THE ACTUAL PRICE PAID BY THE CUSTOMER FOR SUCH PRODUCT. THE PROVISIONS SET FORTH ABOVE STATE SPIRENT'S ENTIRE RESPONSIBILITY AND CUSTOMER'S SOLE AND EXCLUSIVE REMEDY WITH RESPECT TO ANY BREACH OF ANY WARRANTY

What is Jitter?

With the increasing use of delay sensitive voice and video traffic over IP and Ethernet networks, understanding the latency characteristics of these networks and network devices has become more important. There are two key statistics to be measured when characterizing the temporal performance of a network: latency and jitter. Too much latency will render interactive applications such as voice and two-way video unusable because the typical person cannot tolerate an excessive delay in the conversation. However, too much jitter can also render the service unusable

The simple definition of jitter is the change in latency from packet to packet. Applications and end-user devices are designed to tolerate a certain amount of jitter by buffering the data flow and designing the processing algorithms to compensate for small changes in latency from packet to packet. Excessive jitter could cause the buffers to overflow or underflow or it could cause the algorithm to break down. This results in problems such as dropouts in an audio stream or a choppy video display.

Depending on the application, the tolerable amount of jitter will vary, but it is usually assumed to be less than 50ms in most triple play services today. A good service would have jitter of 20ms or less. End-to-end network jitter (or service jitter) can be introduced in a variety of different ways. It may be caused by packets in a flow taking different routes through the network due to congestion or link failure. This type of impairment is temporary and may also result in other problems such as packet loss or out of sequence packets.

The more important cause of jitter that should be understood and measured is the one that will always be present – the jitter introduced by the network devices. The buffering, queuing, and switching architectures of any network device will have an inherent amount of jitter. This inherent jitter could vary with traffic characteristics (packet burst distribution, packet length, traffic priority), traffic load, number of users, device load, etc. As traffic traverses a network, the jitter will be compounded by each device through which it passes.

When designing a network or network service, it is necessary to have an accurate measurement of jitter and to be able to quantify the amount of jitter that may be introduced by the various network components (such as routers and switches). Having high performance devices (high performance = low jitter) in the network will reduce the cumulative jitter and provide high quality services to the users.

There are several ways of measuring jitter. Before discussing how jitter is measured, it first needs to be given a clear definition. The simple definition of jitter is the difference in latency of consecutive packet pairs.

For example, two packets (packets A and B) are sent through a network:
Packet A takes 15 ms to get though the network.
Packet B takes 18 ms to get through the network.
The difference in the latency between the two packets in the pair is 3 ms.
Jitter = | 15 – 18 | = 3 ms.

Four parameters need to be measured to calculate jitter: the transmit time of the first packet in the pair, the receive time of the first packet in the pair, the transmit time of the second packet in the pair, and the receive time of the second packet in the pair. If A is the first packet and B is the second packet, then jitter can be expressed as:

$$|(RxA - TxA) - (RxB - TxB)|$$

Jitter is always expressed as a positive number, so the absolute value of the difference is used. Average jitter is the average value of the jitter of consecutive packet pairs. If the latency is constant and each packet experiences the same delay, then the jitter would be 0 (since the difference in latency from packet to packet would not change). Figure 1 below provides a graphic representation of how jitter is calculated. In this example, two successive packets A and B are transmitted at times TxA and TxB. Packet A takes L_A seconds to get through the network and packet B takes L_B seconds. The jitter of this packet pair is the absolute value of L_A - L_B.

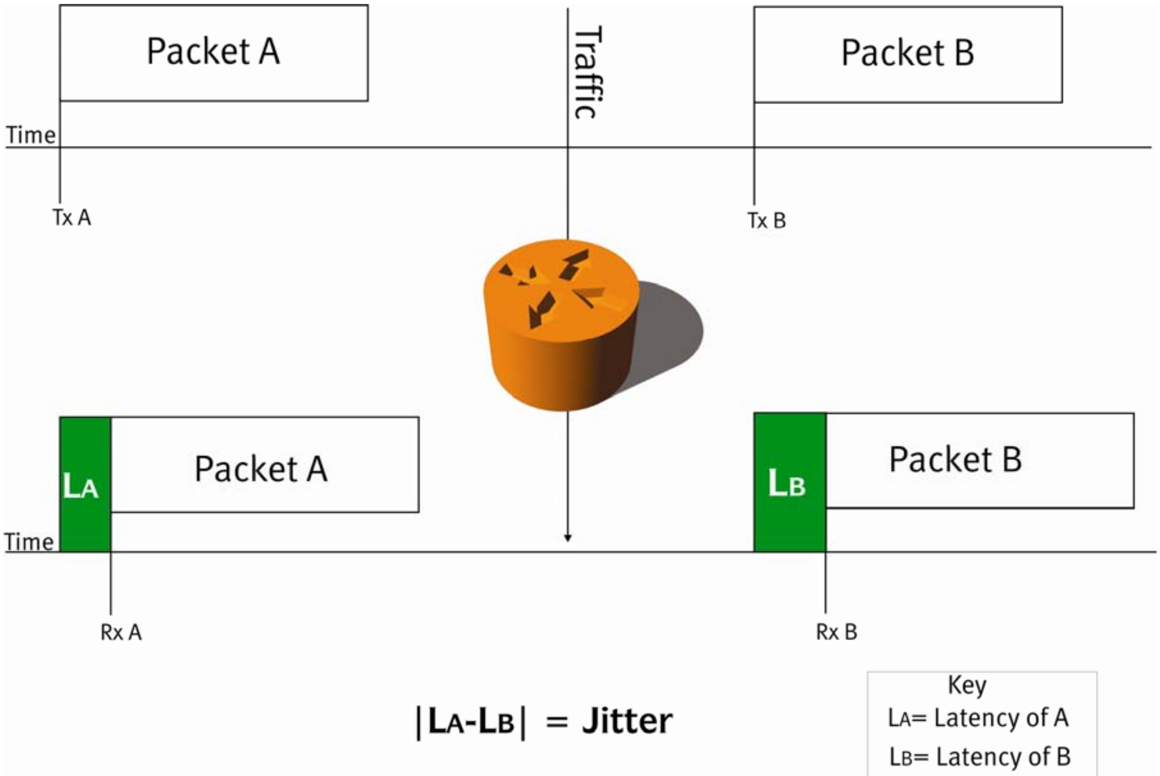


Figure 1

Measuring Jitter

Even though calculating jitter is simple, measuring it is not. There are three common methods used to measure jitter: inter-arrival time method, capture and post-process method, and true real-time jitter measurement method. The hardware required to measure jitter will vary depending upon the measurement method used and data rate to be analyzed. Slow speed traffic can be analyzed using a PC and the capture method. True real-time jitter measurement will need specialized hardware that can accurately process the data in real time at line rates up to 10Gb/s. This paper will examine the advantages and limitations of the three methods. There are other methods of measuring jitter that are not considered in this paper because they are insufficient for laboratory test environments. For example, jitter can be approximated as the difference between the maximum packet latency and minimum packet latency over a given period of time. This does not measure packet pair latency and the results can be corrupted by macro changes in latency (i.e. the latency through a device could steadily increase from 20 ms to 200 ms over the period of the test. The jitter in this case would be calculated to be 180 ms which would be far higher than the actual packet to packet jitter).

Typical Traffic Scenarios
▪ Send traffic at a constant rate (10%, 50%, 100%) using fixed length packets
▪ Send traffic at a constant rate using varying length packets
▪ Send traffic at a varying rate using varying length packets (realistic bursty traffic)
▪ Send traffic-pair configuration without congestion

Table 1

A typical network or network device test plan would include a measurement for jitter performance under various test scenarios. For example, Table 1 lists a few traffic scenarios that might be specified. The method used to measure jitter should be evaluated against the test scenario and traffic requirements.

Inter-arrival Histogram Method

The inter-arrival method is one of the popular ways of making a jitter measurement. This method uses the trick of transmitting the packets at a known constant interval. By using this trick, two of the four needed parameters are pre-determined. Since the packets are transmitted at a known fixed interval, only the inter-arrival time of the received packets needs to be measured. The difference in the inter-arrival time between packets is the packet-to-packet jitter. The inter-arrival values are measured over a period of time and are displayed in a histogram.

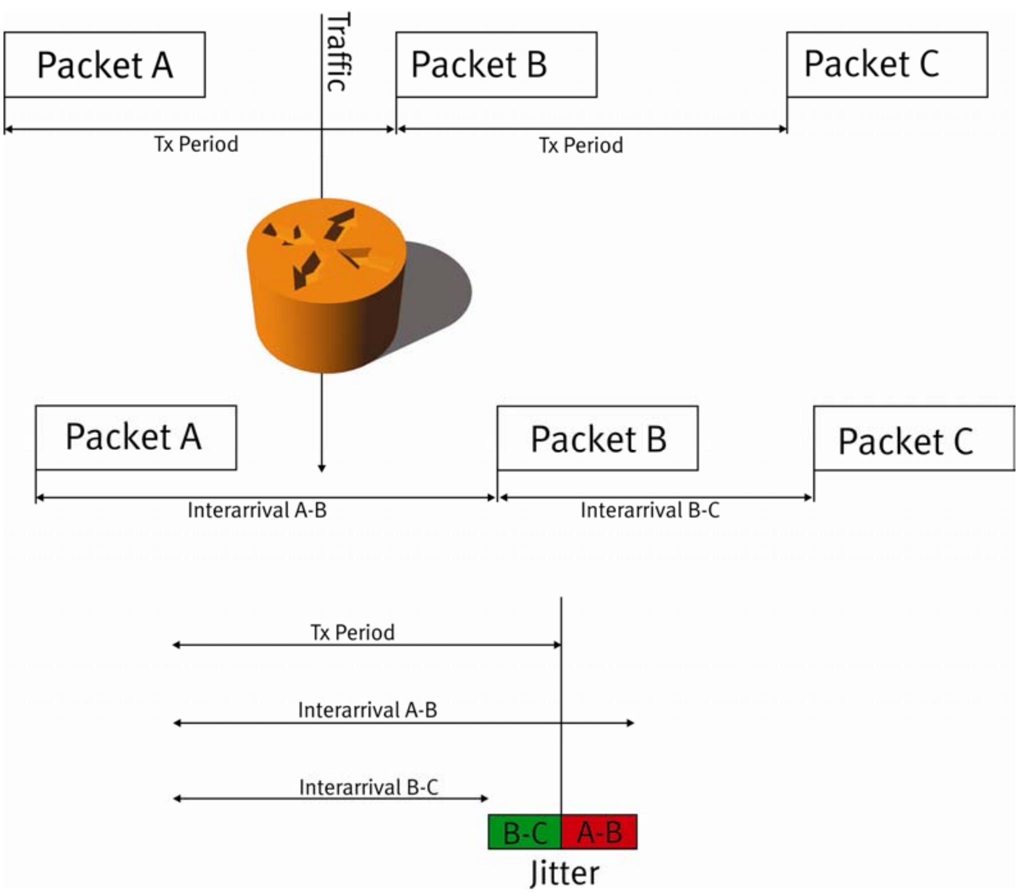


Figure 2

The inter-arrival method has a critical limitation and a few accuracy flaws. The critical limitation is that the packets need to be sent at equal intervals. This restricts the measurement to only constant periodic traffic with fixed packet intervals. Depending on the complexity of the generating hardware, a further restriction requiring fixed packet sizes could also exist (if the hardware can vary packet size but maintain an exact packet-to-packet interval then varying packet sizes can be used). Because the packets must be sent in perfect equal intervals, it is impossible to measure jitter on traffic with a varying rate (bursty traffic).

A key accuracy flaw of the inter-arrival histogram method occurs when a packet is lost (dropped or corrupted). The inter-arrival time between the two packets before and after the dropped packet will be large and will corrupt the inter-arrival histogram. To eliminate corruption of the results, the inter-arrival measurements should be discarded when packet loss occurs. However, most measurement equipment does not have the capability to discard the dropped packet inter-arrival data.

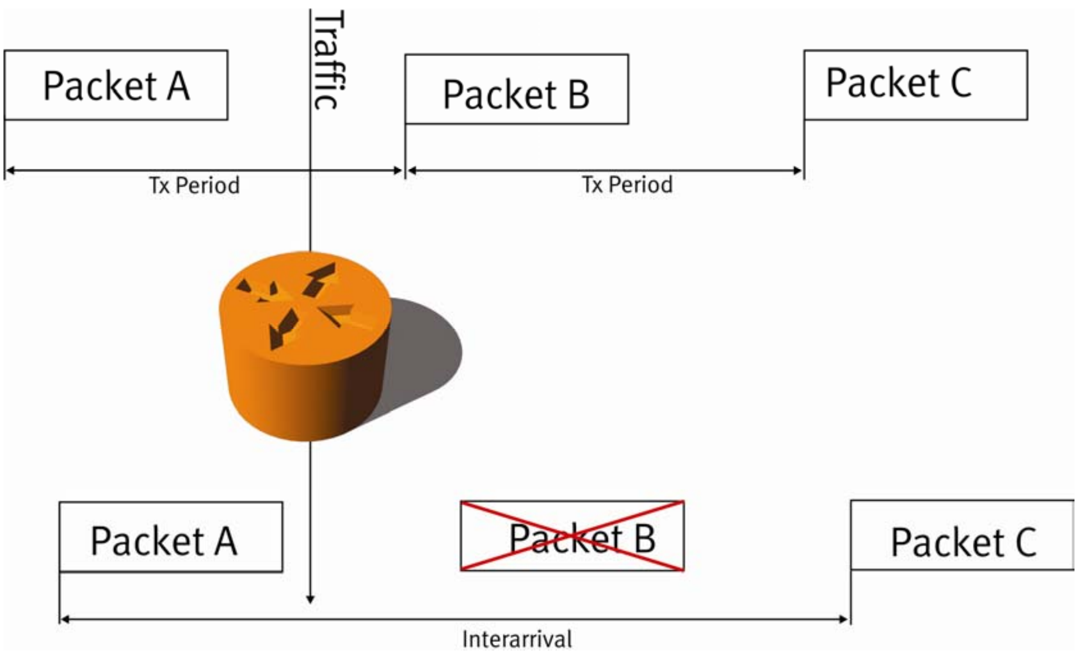


Figure 3. Packet B has been dropped (indicated by the red X) and did not arrive at the destination. The inter-arrival time between Packet A and the next packet received (Packet C) has been calculated incorrectly since this method does not properly account for the lost packet (Packet B). The inter-arrival method will indicate an erroneously high jitter value as a result of these dropped packets.

Another related accuracy flaw present in the inter-arrival histogram method is that it does not take into account packets that may arrive out of order. Packets that arrive in a different order from which they were sent will also corrupt the measurement.

Capture Method

A second common way of measuring jitter is to capture all of the packets and process the data offline. Most test equipment puts a signature in the sent packets and thus the capture file will contain all the needed information (timestamps in the packets indicating the Tx times and the capture buffer hardware indicating the Rx times). The test signatures also include the packet sequencing information so it is possible to compensate for lost or out-of-sequence packets using this method.

The critical limitation of the capture method lies in the fact that the buffer is a finite space and can be filled up very quickly if data is being sent at high speeds. Typical test plans dictate the need to measure jitter over a much longer period of time than is possible with even the largest capture buffers on today's test equipment. Another limitation of the capture method is that it does not allow for real-time cause and effect analysis. Debugging and analysis time can be greatly reduced if the engineer is able to change some traffic load or device configuration parameter and see the feedback in the jitter measurement. This real-time operation is not possible with the capture method.

True Jitter Measurement

To provide a set of industry standard definitions, the Metro Ethernet Forum (MEF) released the MEF 10 specification in 2004. This specification contains a section defining the proper way to measure jitter and takes into account the cases of lost or corrupt packets. Figure 4 shows a flow chart illustrating how Spirent TestCenter 2.0 implements the MEF 10 jitter measurement definition.

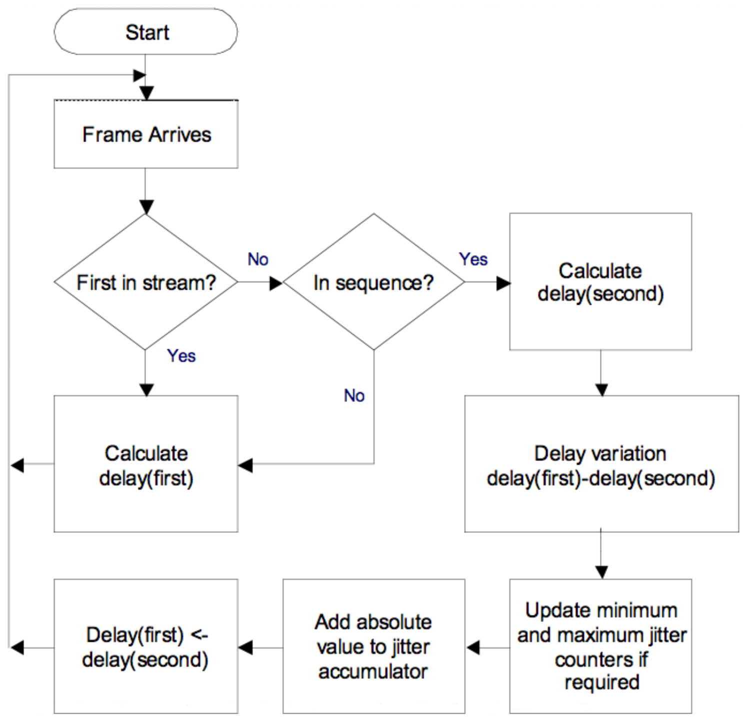
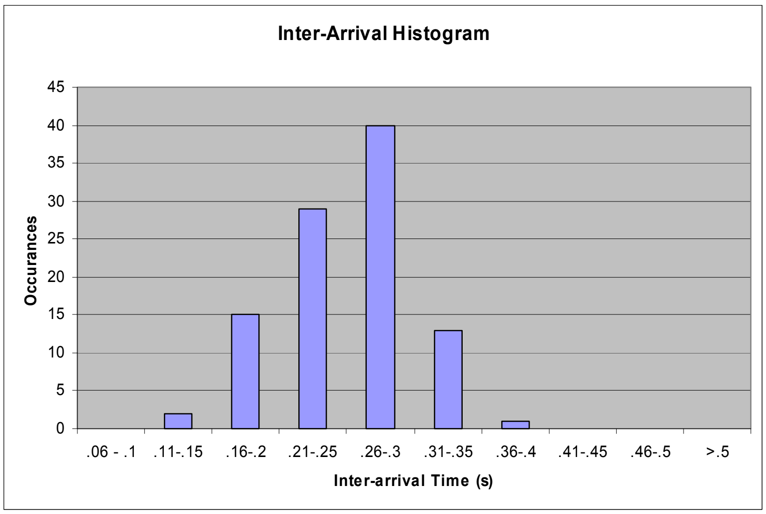


Figure 4

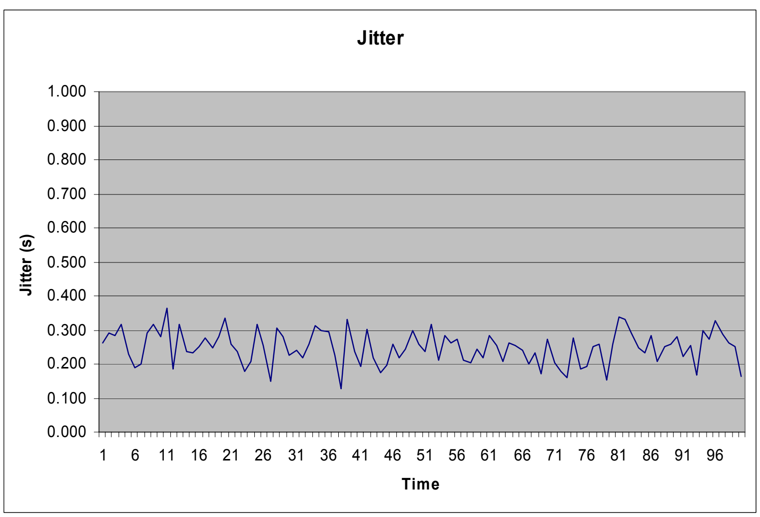
If the packet is the first one sent in the stream then the packet transfer delay (latency) is calculated and stored. If a received packet is not the first packet in the stream then a check needs to be performed to make sure the packet is in the correct sequence. If the packet is not in sequence then the latency results are discarded and this packet is treated as the "new" first packet in the stream. This will eliminate lost or out-of-sequence packets from corrupting the measurement. If the received packet is not the first packet and it is in sequence then the delay is calculated and stored. Next, the delay variation (jitter) is calculated by taking the difference of the delay of the current packet and the delay of the previous packet. The maximum, minimum, and accumulated jitter values are updated and stored. Finally, the delay of the current packet is saved (to be used as the previous packet delay when the next packet that arrives). This algorithm runs in hardware at full line rate up to 10G.

One of the main advantages of the true real-time jitter measurement is that it is not dependant upon having the packets sent with a known interval and can measure jitter on variable rate (bursty) traffic. Also, it is not limited by the size of a capture buffer because the calculation happens in real time as packets are received and does not require packet capture. Finally, it compensates for lost and out-of-sequence packets and produces the results in real-time to allow for instant feedback when varying traffic or device parameters.

Other advantages of true real-time jitter measurement is that it allows more complex analysis views such as jitter charts or jitter histograms. These views produce a much more revealing picture and can substantially reduce test and analysis times. For example, the typical inter-arrival method produces a histogram of inter-arrival times showing how many packets were received in each inter-arrival bucket. With only the inter-arrival histogram, it is not possible to determine when, where or why any abnormal amounts of jitter occurred. All that can be determined is the approximate maximum, minimum and average jitter. Looking at Histogram 1, the jitter (width of the histogram) is measured to be about 250ms. The corresponding line graph of the measured jitter is shown in Graph 1.

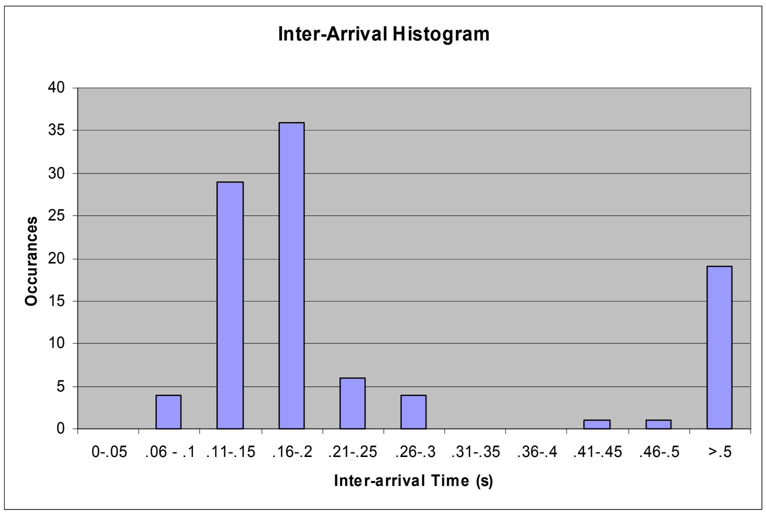


Histogram 1

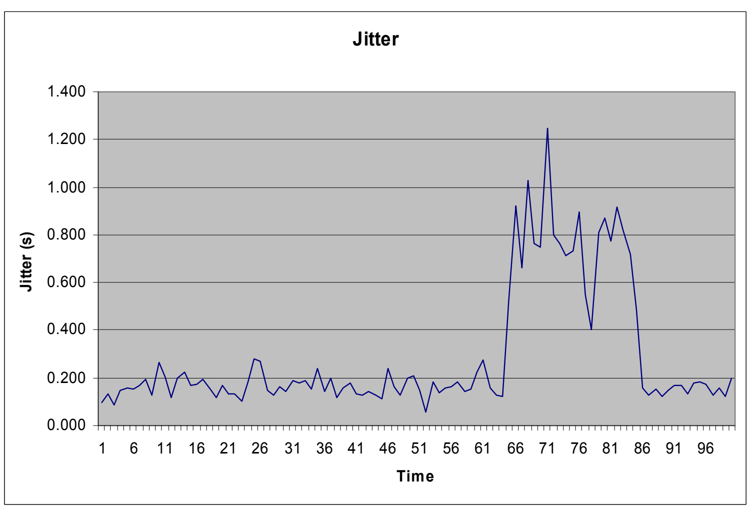


Graph 1

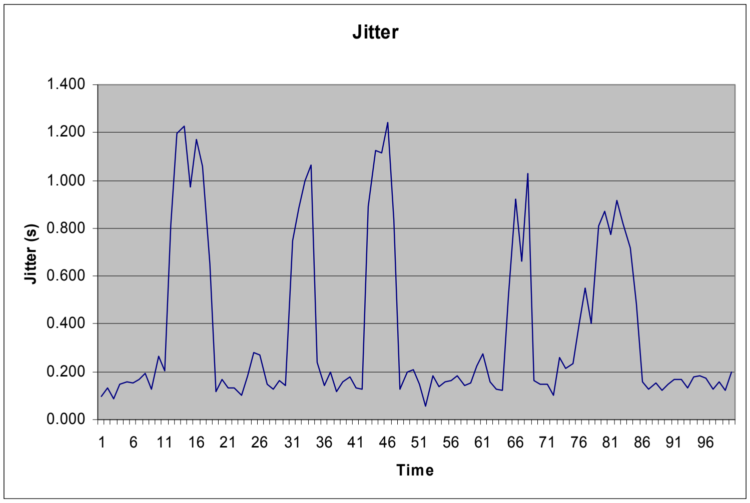
However, Histogram 2 does not provide such a clear picture. Roughly 1/5 of the packets have a much higher inter-arrival time. From the second histogram it is not possible to tell how the jitter was distributed over time. The jitter may have occurred as a single burst (Graph 2A) or it could have been distributed as several bursts of higher jitter (Graph 2B). Both Graph 2A and Graph 2B will produce exactly the same histogram shown in Histogram 2. This complicates the problem analysis and creates a difficult task for the test engineer. If the user was able to see a live graph of jitter vs. time, the problem becomes much clearer and analysis proceeds quickly. Also notice that if the histogram bins are not properly set, the jitter peak is not able to be measured properly and information is lost. For example, the maximum inter-arrival bucket in the example was set to .46-.5 seconds. The inter-arrival information above .5 seconds was lost because the maximum inter-arrival time was exceeded during the test. This would require the engineer to re-run the test (and it may not even be possible if the event is not easily reproduced).



Histogram 2



Graph 2A



Graph 2B

Conclusion

The following table summarizes the key requirements for measuring jitter. It is clearly evident that the true real-time jitter measurement is the superior method and provides test scenario flexibility, accurate results, and real-time analysis capability. With the advancement in the capabilities and performance of state of the art test equipment, this more accurate and powerful jitter measurement method is now available to test engineers. By having an accurate and clear picture of jitter performance, the test engineer can better understand the jitter performance of their network or network device.

Key Requirements for Measuring Jitter

	True Real-time Jitter	Inter-arrival Histogram	Capture
Accommodates all required traffic scenarios	Yes	No	Yes
Able to continuously measure jitter performance over long test periods	Yes	Yes	No
Properly calculates jitter when packet loss occurs	Yes	No	Yes
Provides detailed real-time jitter results (graphs and histograms) and allows efficient problem analysis	Yes	No	No
Able to measure jitter as defined in MEF 10	Yes	No	No

Table 2