FLUKE networks

Guide to Deploying and Troubleshooting Video in the Enterprise

Table of contents

Overview:

Overview

Recently, a study commissioned by Cisco® Systems made a prediction that added a new word to the lexicon of the network engineer and IT director: zetabyte. The study predicted that by 2013, two-thirds of a zetabyte of video traffic would be on corporate networks. Many people quickly researched the term and found that it equates to one billion terabytes of traffic, a staggering number. Also, that same report predicted that 55% of all corporate traffic would be video, a dramatic change in the composition of information on the corporate network from what is currently there. Where will all of this video traffic come from? What forces will cause this? And why should network engineers pay attention?

Most network engineers just treat video as another application on their network and that may be sufficient in the early stages. The business case for corporate video is so strong, however, that, at some point, most network engineers will be responsible for delivering reliable, high-quality video across their network. Unfortunately, the combination of significant bandwidth requirements and the real-time nature of video, means that your network and toolkit may not be up to the task. The first step is understanding video and the second is being prepared to support and troubleshoot it.

Introduction to Video

Video, the dominant component in most streaming applications, is certain to be on our networks more frequently and in greater quantities for many reasons. So, throughout this paper we will interchange, as is done in industry, the terms streaming media, video and the term audio/video. The increase in video traffic will come from:

a) ***The increase in the popularity of video.*** It is easier and less expensive to produce video than it has ever been. For about one hundred dollars you can buy a video capture card and the editing software and begin make movies, editing those that were produced by other sources, and use what you captured with your digital camera, web cam, or wireless phone. You can create the output in formats that vary from high definition to low enough resolution to be viewed on another phone.

b) ***A wide variety of endpoints can capture and play sound and video.*** Phones, cameras, hallway speakers, digital signage, and televisions have all joined the IP network as endpoints.

c) ***Travel costs are up, driving the need for video conferencing (VC).*** Like the broader video industry, conference endpoints scale from high definition room systems with surround sound to desktop clients. The underlying technologies are making it possible to interconnect individuals using both the corporate IP network and the Internet.

d) ***Social networking sites*** such as YouTube and Facebook are being used to share streaming media files for business purposes such as messaging, training, and explaining the assembly and repair of products.

e) ***The line between conventional network television and so-called web TV is blurring.*** Employees on the job and on the road catch up with episodes they've missed by going to sites such as hulu.com. Nearly every major CATV, satellite and telco delivering broadcast TV is doing a pilot project in which popular programs are delivered over the Internet. Hospitals and universities have discovered that by taking a carrier's TV feed and separating the channels, they can encode each on a multicast IP address and deliver the channels to hundreds or even thousands of TVs without using any bulky coax cables.

f) ***Security, surveillance and traffic monitoring*** can be delivered efficiently to multiple viewing sites including mobile devices such as patrol cars.

These are just a few of the influences driving video traffic.

Applications of streaming media are varied but seem to be based on vertical industry and the functional purpose of the streaming. As mentioned above, any organization with clients or students on a campus for more than a few hours should expect them to view video. So video is a necessity for schools, colleges, hospitals and health centers. These campus locations are also ideally suited for IP intercom. For example, a school district can easily extend its building

intercom to the entire district by using its IP data network. Retail store managers and grocers realize that while the customer shops, they can be provided with a video sales promotion on monitors strategically located around the store. Training, both in the corporate world and in education is undergoing a radical change as students can take classes taught by a centralized instructional staff and delivered using the Internet as the backbone network.

It's an exciting time to study and discover the uses for streaming media. However, these new opportunities will also bring new challenges. Among these are:

1) ***Video often consumes vast amounts of bandwidth*** when compared to conventional data applications. While new compression technologies such as H.264 mitigate part of this problem, compressed video can demand from ten to one hundred times the bandwidth that a database query or email might require.

2) ***Video technologies vary considerably in the way that they are transported over IP.*** We haven't had a lot of experience in troubleshooting all of the different forms of video. Some are carried in TCP; others in UDP. Some use HTTP; others use RTP. The vast majority of the training, tool usage and experience with these protocols involved data applications and voice, not streaming media or video.

3) Because of the reasons cited in #2, ***the effect of network problems on video output can vary considerably.*** Sometimes packets with errors that are dropped can be insignificant. With a different form of video, the same error level could be devastating.

Clearly, it is important for us to learn a lot more about the various forms of video in order to be effective in dealing with them in our networks.

Understanding How Streaming Applications Work

Video that is distributed in any system usually goes through a sequence similar to what is shown in Figure 1.

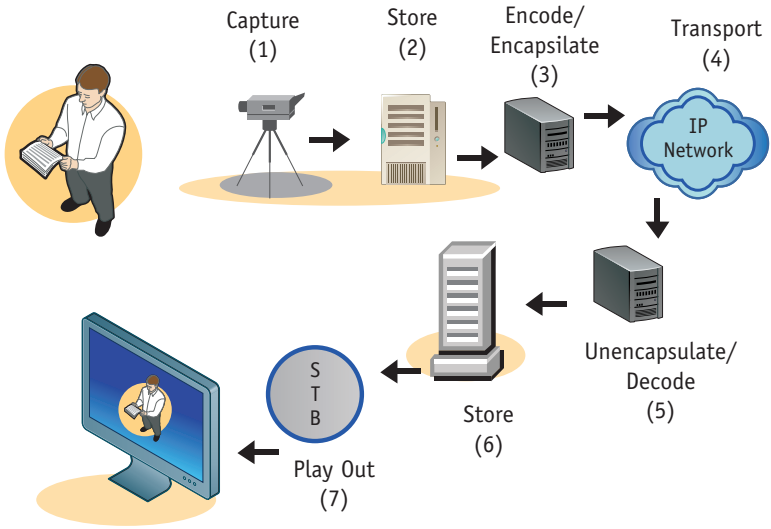

Figure 1:Steps in Video Delivery

In this paper, we intend to explain the essential parts of each step and to take a more detailed look at how the performance of steps 2, 4 and 6 impact step 7.

At the source, if video is captured into a file, it is often referred to as a container. Popular formats for this file include AVI, MPEG, and WMV. If the file is intended for immediate delivery and play out, such as in a live sporting event, the file is encapsulated in a transport stream and delivered in near real-time. In these cases, UDP protocol is used and retransmissions of lost data are not possible. At the destination, the video is buffered very briefly for the purpose of smoothing play out with a set-top box (STB). This delay is usually no more than a few seconds.

The second scenario, and the one that is rapidly becoming popular, has the same essential steps as the first. However, play out is done in the software by a video player that replaces the role of the set-top box. One of the major advantages to this approach is that the video player can use the buffering, decoding and control functions of the computer in which it is imbedded. Sometimes the player is a separate piece of software such as Windows® Media Player, Adobe® Flash® Player or VideoLan's VLC Player. However, there is an increasing tendency for the player to be downloaded with the video file for automatic execution or imbedded in the browser such as Internet Explorer or Firefox. Figure 2 shows the essential steps in this method.

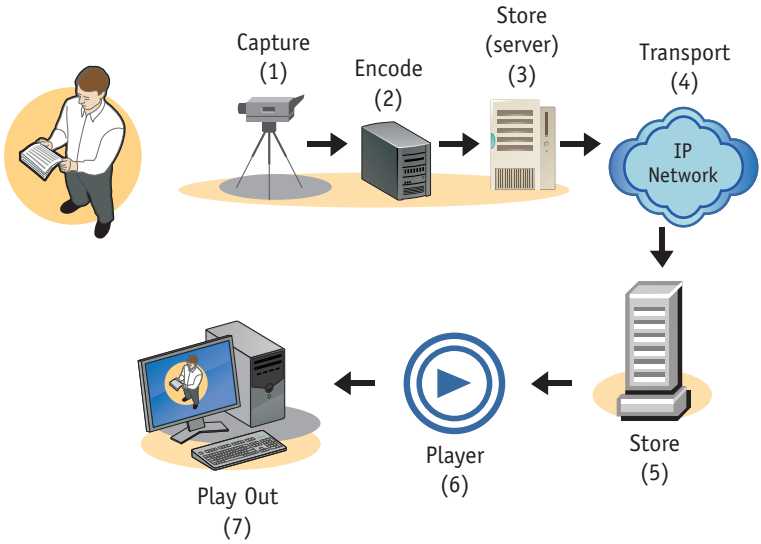

Figure 2: Browser Based Video

Since a video file is transferred from a server to a client PC in this method and played out whenever the player has enough video for presentation, the transfer is almost always a form of TCP based file transfer, similar to when data files are moved using FTP (file transfer protocol). If packets are lost or delayed, retransmission is automatic. Both Microsoft® and Adobe® use this technique. YouTube, Facebook and many sites that offer television episodes often use this method of streaming as well.

The last of the streaming techniques that involves delivery from a source to individual users is shown in Figure 3. What is unique about this method is that the video has been stored at or near to the source and the user who wishes to view it, requests it, and is authenticated by paying a fee. In this case the entire video container file is stored on a server and is encoded/encapsulated for delivery when the source control function grants permission. Pay-per-view (PPV), video on demand (VOD) and content delivery networks (CDN) use this technique. Often TCP/FTP is used to transfer the file from the ***origin*** server to a ***caching*** server near to the potential customer. However, when the video is streamed to the requester, it will use one of the other techniques discussed above involving either IPTV or a browser.

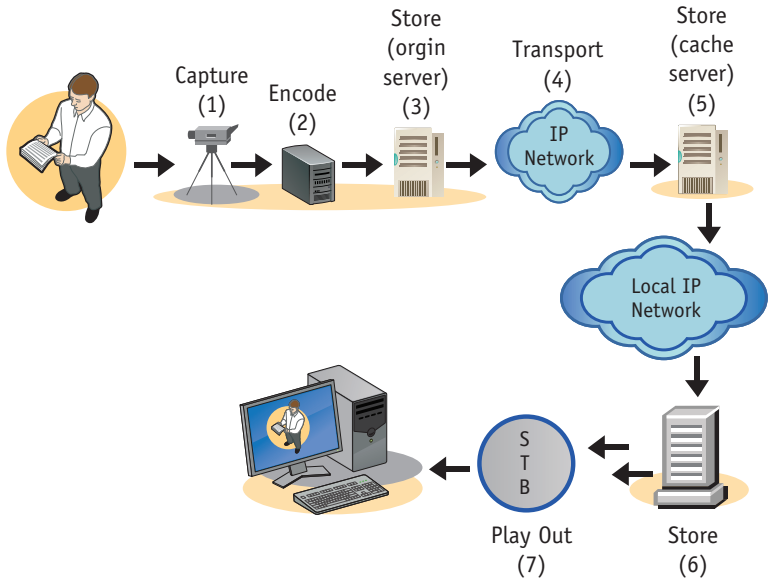

Figure 3: Delayed Video Delivery

The last streaming technique is unique because it generally involves two-way delivery of the video. This is video conferencing (VC), shown in Figure 4.

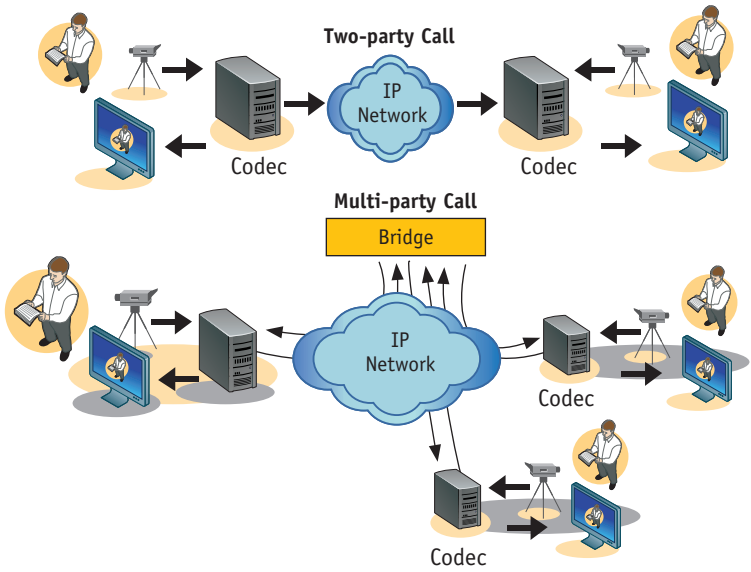

Figure 4: Video Conferencing

For years the VC industry depended on telco circuits such as ISDN and T-1. Today, virtually all VC systems use an IP backbone and many use the Internet. Conferees on a call have cameras and microphones to generate the audio and video signals. However, practically no buffering of these signals takes place at the source and they are compressed, encapsulated in IP and sent immediately. If two parties are conferring, the packets are nearly always carried in UDP using Real-time Transport Protocol (RTP). When the third and successive caller joins, a device comprised of hardware and/or software called a ***bridge*** is used. The audio and video from each source is transferred to the bridge. In the bridge, the signals are combined to create an image that shows two or more of the participants. This combined signal is sent to each participant allowing all conferees to see and to hear the current presenter. Each of the individual source streams and the combined stream are delivered using unicast addressing so the

amount of bandwidth consumed can be considerable. The VC industry has not yet embraced multicast addressing but there are reports that several vendors have products under development that will incorporate it.

The Four Major Types of Video in IP Networks

There are four types of video that are more comon than any other type in IP networks: IPTV, Adobe® Flash®, Microsoft® Silverlight® and video conferencing. We will consider in some detail how each of these works.

IPTV

While the term IPTV is sometimes confused with several forms of video, it is characterized by three distinct things:

1) It uses a delivery mechanism called an ***MPEG transport stream.***

2) It is the technique most frequently used by service providers such as cable companies and telcos to deliver IP video.

3) This form of IP video is much more sensitive to network packet loss than any other form of video. Gradually, though, as we said earlier, enterprise customers are beginning to realize that it has advantages in certain campus settings such as schools and hospitals. We consider it first because it best reveals how video distribution over packet networks function.

At the source, a conventional video signal such as NTSC or DV is fed into an encoder. The encoder separates the video from the one or several audio signals and creates a compressed bit stream for each called an ***elementary stream***. A separate elementary stream is also formed that carries control information. This is shown in Figure 5.

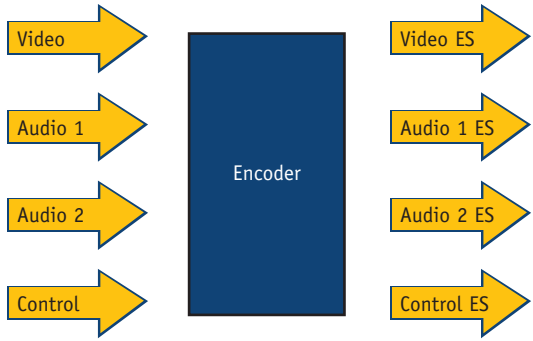

Figure 5: Encoding for IPTV

The compression methods for both the audio and video almost always follow one of two standards, either MPEG-2 or H.264 (MPEG-4, Part 10). These streams are broken into 188 byte blocks called ***transport packets***. Each transport packet contains 184 bytes of data (payload) and a four byte header. These streams are then multiplexed together in to a ***program stream*** by placing several of the transport packets in each IP packet. Generally there are seven transport packets in each IP packet as shown in Figure 6.

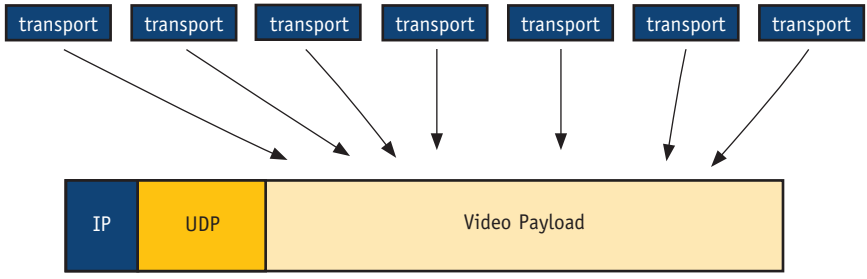

Figure 6: Transport Packets in IP Packets

Notice that the packets do not contain Real-Time Protocol (RTP). Contrary to popular belief, RTP is rarely used in this form of video because adding the header adds little functionality but increases each packets size by 12 bytes. As shown, the total size of each IP packets is then 1358 bytes. When you analyze video traffic on a network, IPTV is often very obvious because of this constant size of the packets. Figure 7 shows a screen from Fluke Network's ClearSight® Analyzer in which this is evident.

Src. Addr	Dst. Addr	Len	Protocol	
192.168.1.54	226.2.4.1	1358	UDP	Source port: 0
172.16.50.2	226.1.4.1	1358	UDP	Source port: 0
192.168.1.57	226.3.2.1	1358	UDP	Source port: 0
192.168.1.57	226.3.1.1	1358	UDP	Source port: 0
192.168.1.56	226.4.2.1	1358	UDP	Source port: 0
192.168.1.57	226.3.4.1	1358	UDP	Source port: 0
172.16.50.2	226.1.3.1	1358	UDP	Source port: 0
192.168.1.54	226.2.3.1	1358	MPEG PES	video-stream
192.168.1.56	226.4.3.1	1358	UDP	Source port: 0
192.168.1.54	226.2.1.1	1358	UDP	Source port: 0
172.16.50.2	226.1.1.1	1358	UDP	Source port: 0

Figure 7: Transport Packets in IP Packets

The frequency with which packets are sent depends on how quickly the respective elementary streams fill the transport packets. As soon as the encoder has enough to make seven blocks of payload, an IP packet is formed. Normally, about 7-10 video transport packets are available for transmission for each audio packet. The fourth elementary stream, the control stream, contains information about the overall structure of the stream (for example, how many channels of audio are present.) This control information is critical to the receiver so that it can separate the various signals and resynchronize them for play out.

At the receiver, the IP packets are received and the audio and video transport packets are separated to recreate each elementary stream. From each stream the video and audio signals are recreated and synchronized. All of these steps are carried out by the set top box (STB), if a TV is used, or by a software player, if the video is being viewed on a computer.

Since the MPEG transport stream is delivered using UDP, retransmissions are impossible. Consequently, the receiver uses a ***jitter buffer***, as is done in VoIP. The jitter buffer is simply memory set aside in the receiver to temporarily store a few packets. This accommodates for the fact that the packets arrive in an irregular manner and possibly out of order. This buffer makes smooth play out possible. There is a major difference between this situation and VoIP. IPTV packets are not sent in the synchronous manner that VoIP packets are sent. It is typical for VoIP packets to be sent every 20 ms. As we said, the spacing of the packets depends on how effectively the video is being compressed at that instant. If the picture is complex or has a lot of motion, the compression is less effective and the packets are sent more often. If the scene has little motion and a simple background, the packets will be sent less often. Therefore managing the jitter buffer in order to keep it partially filled is a critical receiver function. If the buffer overfills, packets can't be stored and are dropped. If the buffer empties, there is nothing to play. In voice, the sound can sometimes be estimated when this latter condition occurs. But video is too complex and such error concealment techniques are not yet developed.

One other element of IPTV is critical. Since the signal is usually coming from a single source and being delivered to a great many destinations simultaneously, ***multicast addressing*** is used. This assures that in the IP backbone, the minimum number of programs are headed downstream on each backbone link. Figure 8 shows this. A program is only carried by a link if there is a downstream viewer of that program.

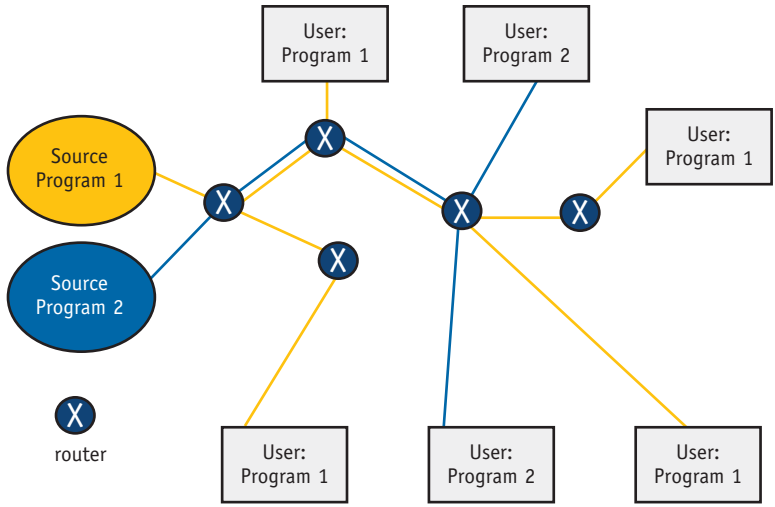

Figure 8: Multicasting Video

Referring back again to Figure 7, you can see that the destination address in packet 1 is 226.2.4.1. Addresses whose first value is between 224 and 239 are used in multicasting.

So, what happens if the user is watching Channel 6 carried on address 224.12.1.6 and changes to Channel 7 which is carried on 224.12.1.7? Following the ***Internet Group Management Protocol (IGMP)***, the user's STB sends a packet called a **leave** upstream to the router. The router stops copying packets with destination address 224.12.1.6 to the user's port. Then the receiving STB sends a second packet called a ***join*** which requests delivery of the packets with destination address 224.12.1.7. The router begins copying those packets to the port to which the user is attached and the program begins to appear on the user's screen. This is sometimes called ***channel zapping***. It is somewhat more involved than described here and is occasionally a source of problems. We'll discuss that in more detail in a later section.

Adobe® Flash®

Various reports have indicated that from one half to 90% of all Windows® computers have Adobe's Flash Player installed. Whatever the exact amount, it is a very common tool used to download streaming media. Consequently, Flash media will continue to comprise a significant amount of bandwidth on corporate networks.

Generally, the audio and video to be streamed are on a server in a container file called an ***.flv*** file. A separate file called an swf file contains all of the information needed to stream the file when the user requests it. The user usually does this by clicking on an item which contains the source address of the swf file. A connection is established and the swf file reads the flv file and starts the streaming to the user. Adobe Flash media operates in two modes: ***streaming*** and ***progressive download***. Streaming a video can only be done from a server running Flash server software. The client can negotiate delivery using low or high bandwidth based on its capabilities. However, once a stream is started, it is played immediately and there is no ability to reverse or pause. It is similar to live TV.

On the other hand, when progressive download mode is invoked, the entire flv file is committed to being transmitted. The client player buffers just enough to begin play out. As more and more of the file is received, it is cached in memory or on disk until it is needed.

On the network, the primary means of controlling delivery of flash video has been to use the Real-Time Messaging Protocol (RTMP). In Figure 9, you can see such an RTMP based transfer.

1	,M	192.168.0.101	96.17.149.72	54	TCP	1735 > 80
2		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
3		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
4		192.168.0.101	96.17.149.72	54	TCP	1735 > 80
5		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
6		192.168.0.101	96.17.149.72	54	TCP	1735 > 80
7		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
8		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
9		192.168.0.101	96.17.149.72	54	TCP	1735 > 80
10		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
11		192.168.0.101	96.17.149.72	54	TCP	1735 > 80
12		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
13		96.17.149.72	192.168.0.101	1514	HTTP	Continuati
14		192.168.0.101	96.17.149.72	54	TCP	1735 > 80

Figure 9: RTMP Transfer

The RTMP Protocol allows for:

- establishing and terminating a connection with the client.
- the client to identify files it would like played.
- identifying a recorded rather than a live stream.
- negotiating audio and video codecs to be used for play out.
- indication of a pause or play in progressive download as well as many other functions.

Recently, Adobe announced support for another method of delivery it calls HTTP tunneling. In this technique, the flv file is simply treated as any object on a web page would be when the HTTP Get command is sent to the server. However, within the downloaded file is an indication that it is a Flash file and the browser passes the file to the Flash player for play out. This is commonly done with videos that are downloaded from YouTube.

Microsoft® Silverlight®

Silverlight is a recent addition to the various streaming technologies. It appears to be quickly gaining in popularity. Microsoft's earliest streaming option was called ***stream thinning.*** It involved lowering the frame rate as network conditions deteriorated. In 2002, Microsoft added ***Intelligent Streaming*** which allowed for adjusting both the frame rate and the bit rate. Unfortunately, the player had to re-buffer frequently to make the change and it was never considered a totally satisfactory technology solution. Also, this method did not support progressive downloads.

Recognizing the desire of the marketplace to choose HTTP based streaming over the various proprietary implementations, Microsoft introduced its Silverlight architecture and called the streaming method ***smooth streaming.*** Both the 2008 Democratic National Convention and the 2008 Beijing Olympic Games were successfully streamed using the Silverlight architecture. The main feature of the architecture is the HTTP adaptive based streaming. While Silverlight supports the RTSP (Real Time Streaming Protocol) used in its earlier implementations, it is clear that the adaptive method of smooth streaming is the preferred method. While RTSP allows for upstream commands to the server to play, to pause and to stop, smooth streaming is a progressive download technology. Once the video begins to stream, the entire file will be delivered and the play, pause, and stop controls are handled by the player. What allows for the adaptive nature of the video stream is the fact that the video is sent in chunks of about 2-4 seconds. These chunks are based on an MPEG transport concept called a ***Group of Pictures (GOP).*** We won't go into detail here about GOPs because that will be important to cover when we discuss troubleshooting video. However, it suffices to say that the GOP idea is a critical part of nearly all video delivery systems and is what makes changing the rate method so easy in smooth streaming.

HTTP adaptive streaming is likely to become even more popular for several reasons:

- Since HTTP uses port 80, firewalls naturally pass the streams. Blocking the videos will be done selectively by content filters. However, the default will be to pass the videos.

- Since the streams are sent in chunks, changing frame rates and bit rates happens transparently as the player chooses which chunks to play.

- Web based technologies such as browser cache operate in a standard manner with these steams.

- The video can be passed to an external player such as Windows Media Player or it can be played by a player embedded in the browser.

Figure 10 shows a Silverlight download in a Fluke Networks' ClearSight Analyzer screen.

Dst. Addr	Len	Protocol	Summary
96.17.149.72	54	TCP	1735 > 80 [ACK] Seq=1 Ack=1 Win=65535 Len=0
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
96.17.149.72	54	TCP	1735 > 80 [ACK] Seq=1 Ack=2921 Win=65535 Len=0
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
96.17.149.72	54	TCP	1735 > 80 [ACK] Seq=1 Ack=4381 Win=65535 Len=0
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
96.17.149.72	54	TCP	1735 > 80 [ACK] Seq=1 Ack=7301 Win=65535 Len=0
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
96.17.149.72	54	TCP	1735 > 80 [ACK] Seq=1 Ack=8761 Win=65535 Len=0
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
96.17.149.72	54	TCP	1735 > 80 [ACK] Seq=1 Ack=11681 Win=65535 Len=0
192.168.0.101	1514	HTTP	Continuation or non-HTTP traffic
96.17.149.72	54	TCP	1735 > 80 [ACK] Seq=1 Ack=13141 Win=65535 Len=0

Figure 10: Silverlight Download

As you can see, it appears to be a web transfer using HTTP. In fact, it looks rather like any other transfer. However, within the HTTP payloads is an MPEG transport stream as discussed under the IPTV section. So, here we see the merging of the effort of Microsoft and the telecommunications industry. It becomes clear that our TV's may have IP addresses and it will be hard to tell if we are watching conventional broadcast programs or something being streamed from a web server.

Video Conferencing

Throughout this paper, we will treat systems referred to as ***telepresence*** systems with the term video conferencing systems. We do this because they behave in much the same manner from a network performance standpoint. However, as you will see in some of our examples, they do generally use considerably more bandwidth. Unlike the other video types, video conferencing has two critical requirements:

1) The video must be symmetrically passed between all endpoints.

2) There cannot be more than about one half to one second of delay between the source and the destination.

Most video conferencing systems followed one of the ITU standards in the H.260 family, either H.261 or H.263 to compress the video. Before IP was introduced, video conferencing was very proprietary in nature and required expensive leased circuits from the telephone company. But with the introduction of IP to the conferencing networks, almost everything changed. While H.263 codes were still used, the vendors began to support the idea of H.264 compression. This would bring them in line with the rest of the video industry in using a standard packet format and standard compression technologies. The typical structure of these video packets is shown in Figure 11.

Figure 11: A Video Conferencing Packet

Notice that UDP and RTP are used. This is usually the case in video conferencing. And, like most other forms of streaming, the rate of transmitting packets will increase when the image contains a lot of motion and complexity in brightness or color. As in VoIP, because the UDP/RTP combination is being used, a jitter buffer must be carefully managed to assure smooth play out. In this case, the level of jitter in the network is a significant factor in the ability of the endpoint to manage the video. Too much jitter causes the jitter buffers to overfill and drop packets or to empty. This results in distorted pictures or no video.

The most significant trends in video conferencing have been the integration of desktop clients and room systems, and the introduction of large telepresence systems. While it remains to be seen whether both of these will be sustained in the market, there is an underlying technology change happening that is very important and occurring rather quietly. Earlier, we mentioned that the video conferencing industry was beginning to embrace H.264 compression. The commonly implemented version of this is actually called H.264 AVC (advanced video coding.) Unfortunately systems that use H.264 AVC have some difficulty changing bit rates to accommodate network deterioration. Consequently, they clearly are inappropriate to use when the backbone network is the Internet. This is the basis on which a new breed of VC devices is being developed using H.264 SVC. In this method, the video components are separated into two or more ***layers,*** each layer supporting a higher level of video resolution. Unfortunately, the vendors now supporting SVC are using the typical UDP/RTP structure so that troubleshooting it will be guess work in the foreseeable future.

Deploying and Troubleshooting Video Conferencing Systems

In the first half of this paper we considered the major types of IP video: MPEG transport, video conferencing, Flash and Silverlight. Now we want to focus on how you can successfully deploy video conferencing. We are concentrating on this particular form of video for four reasons:

- According to Cisco's latest predictions, video communications will increase seven times by 2014 (Cisco Visual Networking Index: Forecast and Methodology, 2009-2014; June, 2010).
- Currently, the typical standards writing organizations have not set standards to address network characteristics required for transporting video conferencing traffic. However, the major manufacturers have published their expectations for customer networks.
- MPEG transport video is expected to remain primarily a technology used by service providers rather than by enterprise customers.
- Flash and Silverlight are expected to be transported using TCP/HTTP. Consequently, they can be analyzed and monitored with tools designed to study web-based applications.

There are standards bodies working on guidelines for transporting the various forms of video. As those guidelines become available, Fluke Networks is committed to develop and provide analyzing and monitoring products to facilitate the deployment and maintain the quality of video traffic in compliance with those guidelines.

We will consider these aspects of deploying a video conferencing system:

- Baselining the network
- Monitoring and troubleshooting

Baselining the Network

The most significant considerations in the successful transport of telepresence flows are whether or not there is sufficient bandwidth and if any other traffic will interfere with the flows. Telepresence systems generally exhibit one of two characteristics. Either:

1) The codec gathers enough data to fill an IP packet and immediately sends it. In this case, packets are sent in an irregular manner. When the scene has a great deal of motion or significant variation in color or intensity, the packets will fill quickly and be sent frequently. When the scene is still or has little variability in intensity or color, packets will be sent less often. This pattern of irregular transmission time will vary even within one frame because it takes many IP packets to carry a single frame.

2) The codec transmits the IP packets in a synchronous manner, filling each frame with the same size block of video or audio data. This is done in the manner that we described in the first half when we discussed MPEG transport. Therefore, the timing of the transmission of the IP packets is very similar to the method used in VoIP.

Since both techniques require that sufficient audio and video data arrive without delay, these flows are very sensitive to interfering traffic. TCP bulk transfers of data are the biggest threat. This is due to the fact that the TCP algorithm was written to estimate the amount of bandwidth that is available and use as

much as possible. For example, if you are using a broadband connection from a cable company with a 6 Mbps and use FTP to download a large file, TCP will cause FTP to use as much as 5 Mbps. During that transfer, little bandwidth is available for any other applications. Also, most file transfers use TCP transfer packets with 1460 bytes in each packet. As these packets pass through a router from one interface to the next interface, no other traffic can be on that path. Consequently, such large packets will increase jitter into the flow of any video using the same path.

As a result of these factors, we should measure and record each of the following values at key points in the path the video conferencing stream will follow:

1) Speed and utilization of the link.
2) The amount of TCP versus UDP traffic on the link.
3) Use of the link by any critical business applications.

Let's consider an example. Suppose we have a network with the architecture shown in Figure 12. Remote Office X, where telepresence system TA will be located, is connected to the headquarters (HQ) by a VPN connection. Telepresence system TB will be provided to an executive for home use and will use a broadband VPN connection to the headquarters. Telepresence system TC will be located near the data center in the Corporate HQ. Finally, telepresence system TD will be in a second remote office that has a fiber link connection to the headquarters. A telepresence bridge is also located in the headquarters.

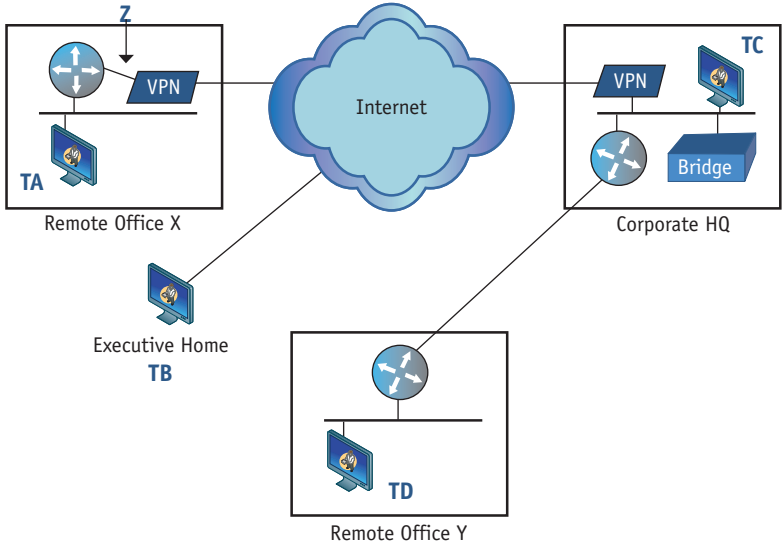

Figure 12: Hypothetical Network Configuration

Suppose you wish to baseline this network for deployment of the telepresence systems. We should know the utilization, composition of traffic and critical business applications on the following links:

1) The Remote Office X LAN.
2) The Corporate HQ LAN.
3) The Remote Office Y LAN
4) The executive home location.
5) The VPN connection between Remote Office X and the headquarters.
6) The link between headquarters and Remote Office Y.

We will use Fluke Networks' ClearSight Analyzer (CSA) in order to gather our assessment data. We can visit each location and connect to the physical links listed above. By capturing data from each segment, we can obtain the values we need. Figure 13 shows a representative screen displaying that DNS, HTTP, and Generic (unclassified) data appeared on the link. To get to this screen, we clicked on the Flows tab.

Figure 13: Summary of Traffic

Next we click on the network label on the left, the Chart tab, and the IP Protocol radio button to see a chart similar to Figure 14. In this instance, almost all of the traffic is TCP rather than UDP.

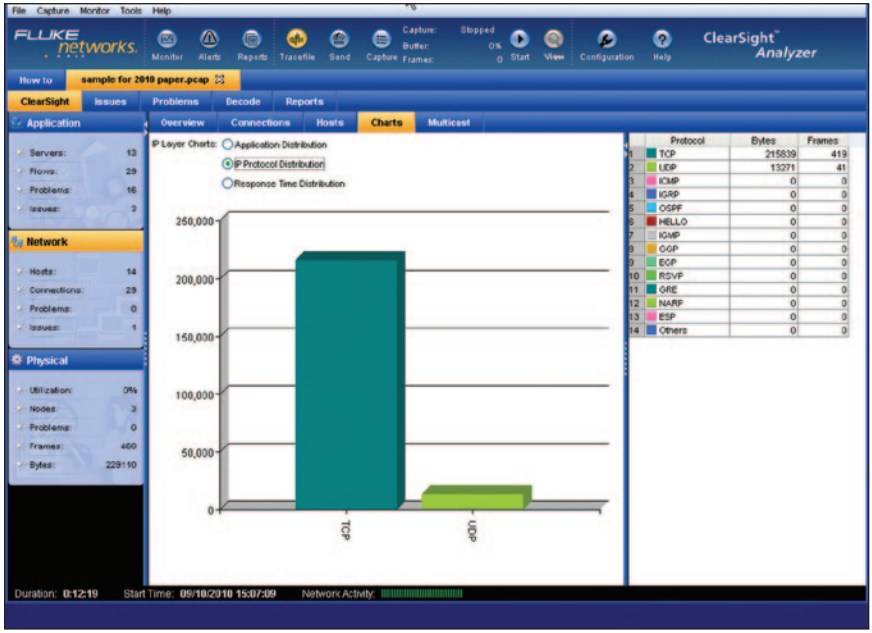

Figure 14: TCP vs. UDP Packets

Finally, we click on the Physical label on the left to get an idea of the utilization of the link. This is shown in Figure 15.

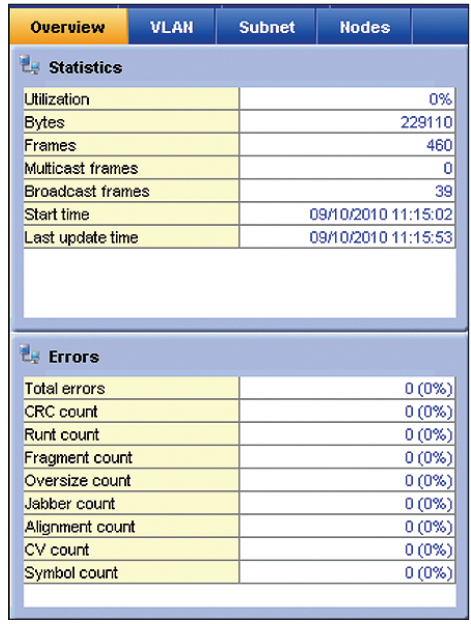

Figure 15: Link Utilization

Fortunately, the utilization is very low. Therefore, since this is a 100 Mbps link, there is little likelihood that the TCP data will interfere with a telepresence flow. Such a flow might use a few megabits/second of bandwidth.

Table 1 shows a representative assessment chart that might be developed for our hypothetical network.

Telepresence Deployment Baseline Data			
Capture Point	**Link Speed Mbps**	**Average Utilization**	**% TCP**
TA	100	20%	85%
Z	50*	45%	75%
TB	20	47%	70%
BR	1,000	22%	70%
TC	1,000	22%	70%
TD	100	40%	50%

* Speed of the VPN link.

Table 1

Since utilization of unit TA is only 20%, it is highly unlikely that it will have difficulty in sustaining a telepresence demand for 4 or 5 Mbps. The measurements at Point Z are aimed at baselining the VPN connection between Office X and the headquarters. Since 50 Mbps are available and 75% of the traffic is TCP, it is reasonable to assume that the TCP traffic might fluctuate for short periods to two or three times the average load. Therefore, we might anticipate occasional periods where the TCP traffic would peak at 20-25 Mbps. This would likely increase the jitter in the telepresence flow but probably would not stop the flow altogether. As you gather this data, keep in mind that the bursty nature of TCP traffic is also dependent on the number of data sources. If two devices are transmitting TCP on a link, utilization will likely fluctuate significantly. However, when twenty five devices are sending TCP data on a link, the utilization will be much higher but the fluctuations will be spread out more evenly.

Troubleshooting Telepresence

Some of the common problems that occur with video conferencing systems include:

1) Failure to set up a call correctly.
2) Failure of the endpoints to negotiate parameters for the call.
3) Excessive jitter.
4) Excessive bandwidth consumption.

A video protocol aware monitoring solution, ClearSight Analyzer can track the call setup process and provide alerts and analysis. Commonly used signaling protocol includes H.323 and SIP. Figure 16 shows a ladder diagram of packets when a call is set-up under the H.323 protocol.

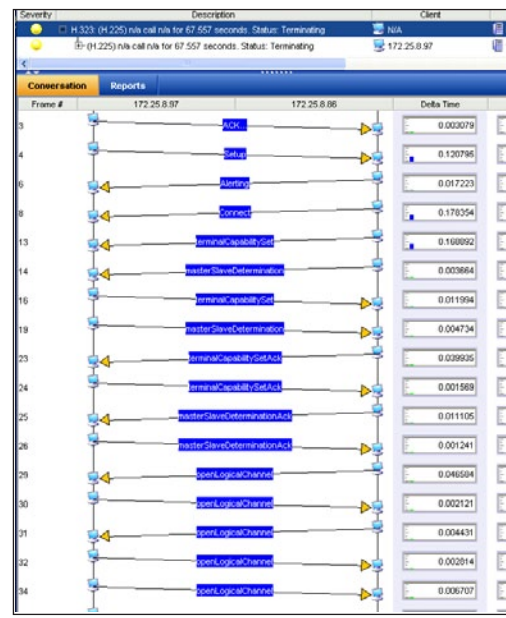

Figure 16: H.323 Call Set-up

H.323 uses TCP. So, the three-way handshake begins the sequence. The Set-up, Alert, and Connect correspond to ringing and answer the call. Next, one of the devices is determined to play the master in the relationship. Then, capabilities are exchanged. These include what codecs will be used, what resolutions and frame rate will be used, and so forth. Finally, Open Logical Channel commands are sent and that creates the ports for the actual flow of the audio and media streams. By studying this pattern, you can determine whether:

1) The set-up session is started.
2) The receiver is contacted and answers.
3) The two end points agree on operational parameters.
4) The port numbers have been defined for the RTP flows.

If everything appears to be happening correctly, the problem might be a result of a particular port being blocked by a firewall. By clicking on the Open Logical label, CSA will open the packet (#29) corresponding to that command. In Figure 17, you can see in the H.245 part of the packet that the port selected is 49179. In the H.323 protocol, the port is called a tsap identifier. Now, you can verify that this port is not being blocked anywhere along the path.

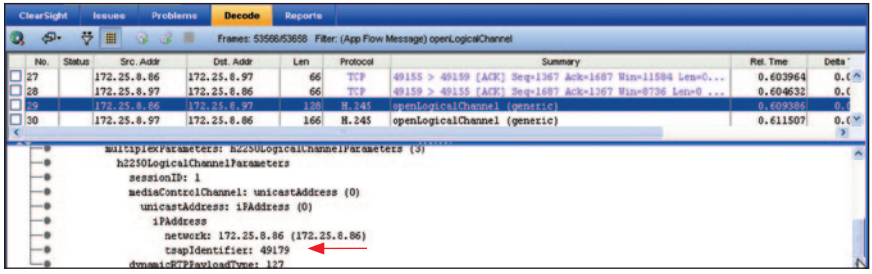

Figure 17: The Port Number

If the call begins but the video or audio is of poor quality, there may be excessive jitter. You can attach the ClearSight Analyzer at each critical point in the path between the caller and the called party. Jitter always increases as the flow proceeds. Also, the jitter might start at a level above zero because the source codec sends the video or audio in a nonsynchronous manner.

As an example, suppose we are having trouble with a call from A in Figure 12 to TD. As we measure the jitter in Remote Office X, in the headquarters, and then in Remote Office Y, the jitter should increase in the flow from TA to TB but decrease in the flow from TB to TA. If the value changes significantly as we move from one location to the next, we have found the source of the jitter. For example, if the jitter increases significantly as we move from Remote Office X to the headquarters, the jitter is likely being caused by traffic interfering on the VPN circuit. The ClearSight screen in Figure 18 shows the jitter value.

A similar analysis can be used to find the source of packet loss.

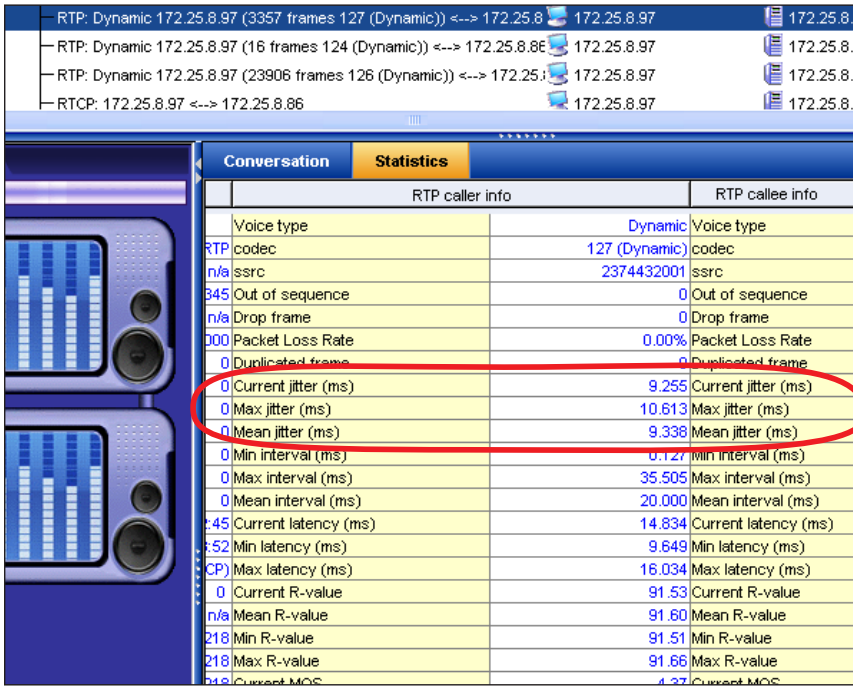

Figure 18: Complete call statistics including jitter measurements

To determine if a particular flow is consuming an excessive amount of bandwidth, we can use the same statistics screen that we used to get the jitter values. The bandwidth used by an audio stream is shown in Figure 19 while Figure 20 shows the bandwidth used by a video stream, labeled “Transaction Throughput.” used by a video stream. They are labeled “Transaction Throughput.”

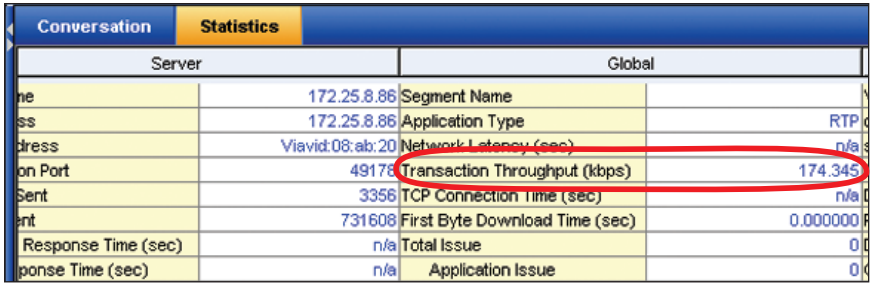

Figure 19: Audio Stream Throughput

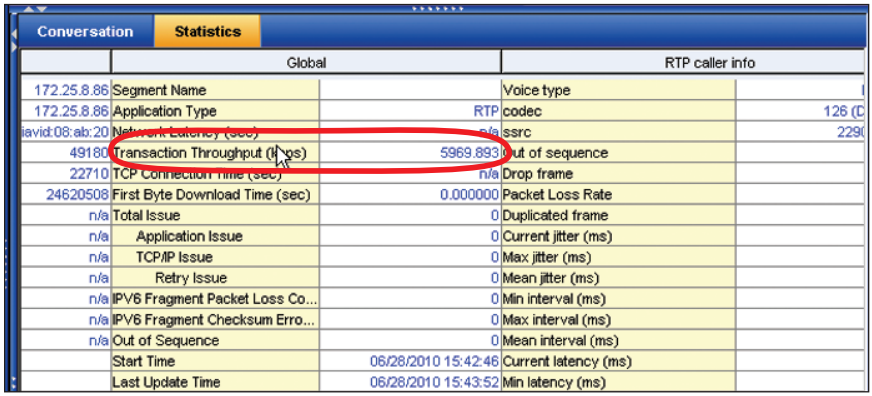

Figure 20: Video Stream Throughput

Using a Network Forensic Analyzer for monitoring and troubleshooting

As mentioned throughout this guide, video traffic requires higher network bandwidth, low latency, and sustainable throughput. When a video stream co-exists with other traffic, data, voice or video alike, and competition for bandwidth occurs, there is opportunity for problem identification and network capacity planning for quality application delivery. A Network Forensic Analyzer, such as the Fluke Networks' Network Time Machine™, has the ability to continually store all traffic for a long period of time and quickly extract and analyze video and data applications simply by specifying the time period of interest. This allows nagging intermittent video feed or application quality problems to be identified and resolved.

The Network Time Machine is video and voice traffic-aware with support for most commonly used video and audio codec. This allows play back of the video and voice stream, and offers video and audio codec sensitive quality metric such as R-Factor for VoIP and VQ factor Video stream. Together with its built-in application-aware analysis function, it can help troubleshoot network problems where video, voice and data traffic co-exist.

Summary

Since the first network was deployed, network engineers have been hit by wave after wave of new technologies. Video is one of the next big ones. How it will eventually impact your network is impossible to predict at this time. But knowing that it is coming can give you time to prepare.

You've already taken the first step – learning about the technology and the ways in which it can affect your network. The second step is to outfit yourself with the tools necessary to understand how your network will impact video and how video will impact your network. The good news is that these tools can also help you manage your network for all applications and users – not just video.

Fluke Networks is committed to helping network professionals prepare for video and any other technologies headed their way. As standards and best practices for networking change, Fluke Networks will continue to introduce capabilities for its tools to maintain and support them as well as keeping you informed of these changes. To learn more about making sure your network is ready for the next wave, visit ***www.flukenetworks.com/enterprise-video***.

Resource Links:

Video in the Enterprise Resource Center:
www.flukenetworks.com/enterprise-video

Download Video Solutions Product Flyer:
www.flukenetworks.com/video-solutions

Download ClearSight Analyzer Software Trial:
www.flukenetworks.com/csatrial

Check out recent blog postings:
The Decorder Blog: **www.flukenetworks.com/decoder**

Fluke Networks
P.O. Box 777, Everett, WA USA 98206-0777

Fluke Networks operates in more than 50 countries worldwide. To find your local office contact details, go to **www.flukenetworks.com/contact**.

Printed in U.S.A. 11/2012 3936305B