

GRAPHIC EXCHANGE

Watch the Maestro

JOHN WARNOCK

Q&A with the
Father of PostScript

FRANK CASANOVA

On Apple QuickTime 4

WIRED SPRITES

QuickTime Report

STOCK OPTIONS

Royalty-Free Market Explodes

ALL PAGES ARE NOT CREATED EQUAL

Fundamentals for Corporate Communications

Quark Pages That Won't Print Right

Tips to Building Better Pages

A Guide to Graphics Courses in Canada

REVIEWS

Markzware FlightCheck

Extensis PreFlight Pro

Macromedia Fireworks 2

Adobe GoLive 4



\$4.50 JUN/JUL 1999



a *QuarkXPress*- or *Director*-like interface. Most of us can produce some of the individual raw materials for the QuickTime tracks — still images, video, audio, *Flash* animations, etc. But to package them up for use on the Internet effectively, we are still dealing with separate QuickTime .mov files, GIFs and JPEG images.

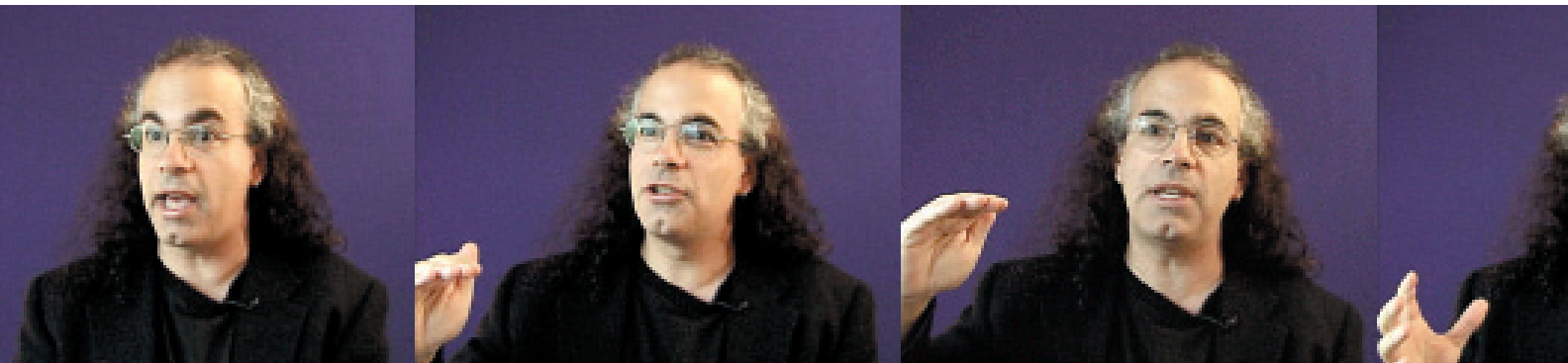
Apple needs to produce an application that controls all the functions of QuickTime assets, a sort of drag-and-drop Web page tool that allows you to:

- import a compressed JPEG file to use as a background
- place QTVR movies on top of that JPEG file
- place digital videos on top of QTVR movies and on JPEG backgrounds

- place *Flash* animations beside the digital movies
- add sound to the production
- add MIDI sound data
- create hot spots for user-controlled interactivity
- link to other Web pages within the same page
- produce Java applets for database, cgi, etc.
- allow for transitions between VR objects, panoramas and videos

Finally, and most important, it should allow you to export a hinted, self-contained QuickTime 4 movie, ready for uploading to an Apple Streaming Media Server.

Imagine a Web page made entirely of one streaming QuickTime movie. HTML would only be used to download



A Frank (Casanova) Conversation about Apple's QuickTime 4

When Frank Casanova, Apple Computer's Director of Product Marketing for QuickTime, talks about his favorite subject, he exudes tremendous exuberance for his product. Casanova has been with Apple on and off for about ten years, first with the CPU group, then the research group. Bob Connolly caught up with Frank at the June newMedia 99 show in Toronto, just days before the official launch of QuickTime 4.0. It was a perfect opportunity to go into some depth about what QuickTime is, what it can do and how to use it.

Bob: Let's start with the differences between QuickTime 3 and QuickTime 4. Can you sum that up for us?

Frank: The big difference between QuickTime 3 and 4 is real-time streaming. It's important to understand the differences between http streaming—which is in essence a file download—and real-time streaming, which is audio and video coming real-time to your desktop. QuickTime 3 supported streaming but it was called http streaming, which was basically where you click on a file server and those bits are transferred from the server to your local hard disk. The cool thing was that you could watch as you downloaded. If your connection

speed was faster than the playback, you had a great experience. If not, sometimes you had to wait for more information to download, and hit Play again, and watch that bit that downloaded. By the end of the download, you ended up with a file on your hard disk. That's cool, because then you could play around with it, play it back, and do whatever you want—except what if the content provider doesn't want you to do that? As we know, a lot of people in the industry are very, very strict with what you can do with their content. Then we come to QuickTime 4, which allows real-time streaming of audio and video over the Internet. A real-time stream sends you nothing more than audio and video, but at the end of the session, you are left with nothing but a good feeling. You don't have the bits, you can't repurpose any of the content; so intellectual property people—content providers—are really excited about real-time streaming, because it doesn't leave anything there at the end. The essence of QuickTime 4 is real-time streaming and a whole bunch of other stuff that goes with that.

Bob: What changes will we see with material that's used? What if it's not put over a streaming server? What are we going to see in per-

formance?

Frank: You'll see differences in quality and performance. QuickTime 3 came out about a year and a month ago. Since then we've been working directly with our codec (compressor-decompressor) vendors — companies such as Sorenson Video, Q-Design, a Vancouver company for our music audio, and Qualcomm for our speech audio. We've worked with these guys aggressively, with our engineering group and their engineering groups, to evolve state-of-the-art codecs over the course of a whole year—and you can do an awful lot in a year. Understand, too, that the performance of Apple's hardware has gone from 233 MHz for an iMac up to 333 MHz; 100 new MHz can do an awful lot of goodness to a codec. Compressing and decompressing video is incredibly performance-oriented, so that the more performance you can throw at it, the better things look. And our codecs have evolved to go with that performance—and QuickTime scales! QuickTime is incredibly scalable—the faster your connection, the faster your computer, regardless of format, QuickTime will scale to fill up as much performance as you can possibly have.

Bob: I noticed that the procedures to download

the page into your browser. The QuickTime plug-in streams the data from the server to your computer, and instead of tiled images, full page JPEG background graphics in millions of colors are transitioned by QuickTime effect generators as animated slide shows. Virtual Reality Panoramas, Object Movies and Digital Videos play in position on top of the background. Interactive hot spots allow for Internet links and navigation of virtual tours. The touch of a button over live video triggers animations, video and sound. Tiny MIDI data files play music that is generated by your computer. Java applets provide database fields for ordering products.

Sound far fetched?

All these capabilities are built into QuickTime 4. But at this point they remain untapped because there are no sim-

ple (key word — *simple*) authoring applications that can get into the guts of QuickTime and present it to us with a friendly user interface.

Apple has just recently ventured into the world of digital video editing software with *Final Cut Pro* and Adobe is screaming murder — *Premiere* is a goner!

Apple must move away from its Apple Media Tool CD-ROM interactive software and release a QuickTime Wired Sprite Authoring tool.

There are, however, several small developers who have seen this opportunity and have built helper applications using wired sprite QuickTime technology. Since we have been promoting our GRAPHIC EXCHANGE QTVR contest, I'd like to introduce you to some third party tools that will come in



QuickTime 4 or the upgrade to QuickTime 4 have been drastically altered. Could you explain that?

Frank: The install process is very different for QuickTime 4 than it was for Quick-

Time 3. There are a couple of reasons why. QuickTime 3 was about seven megabytes in size. We noticed that a lot of people started the download and then stopped. It could be for a lot of reasons—maybe call waiting clicked and they lost their modem connection, or maybe they were using a 28.8 modem and they got tired of waiting for a 7MB file to download. Regardless of which, we had millions of downloads of QuickTime 3. In QuickTime 4, we decided to break it up into three different sections. The final version that ships will have three different types of installs. There will be a minimum install, which is about 2.5MB, there is a full install, which is about 5.7MB, and then there is a custom install, where if you take it all together is upwards of 8MB. Basically you can pick and choose how much stuff you need, versus how much time you have to wait right in the installer. So, if you pick the 2.5MB install, you get full support for audio and video using our best codecs—MP3 support, Flash, wired sprites, and QuickTime VR support are all right in the minimum install. If you're surfing on the Web, and you've bumped into a piece of media that your QuickTime doesn't know, we're not gonna leave you hanging. We're gonna pop up a message that says we've found some media that your current installation doesn't understand—would you like us to go get it? You click 'yes', and your installer will connect with our servers

in Cupertino and download the pieces of software that you need, install it on the fly and let you interact with that media.

Bob: What type of media would that be?

Frank: 3D would be an example. If you want to look at some 3D in a QuickTime movie, the minimum install does not install the 3D graphics viewers.

Bob: How will QuickTime know you need 3D?

Frank: Because it's QuickTime. It found something it didn't understand. The QuickTime movie will ask your player to look at it and your player will recognize that it doesn't have this necessary piece of software and it will signal your installer to flag you that 3D is required.

Bob: How does it accomplish that? By looking at the extensions?

Frank: Yes, it knows what you have, and it sees what it's being asked for. It's all QuickTime!

Bob: So, it analyzes your whole computer?

Frank: Well, it knows what version of QuickTime you had installed and it knows at all times where you are, what you have and what you want to have with this piece of media you just ran into. In addition, every week or so, when you connect to the Internet, your QuickTime application will go back to Cupertino, and it will compare your bits to our bits, and if there are some new ones on our site, we'll ask you, do you want to get them right now? Yes or no? As a matter of fact, there's also a way to force the installer to go check your bits against our bits. We can add updates all the time now. We don't have to wait for big monolithic, huge introductions. With QuickTime 4 and its intelligent installer, if we find something we want to fix, or if there's a new piece of software we want to have qualified, or if there's a new button for the channel drawer, we can just put that on our site, and it will change the revision of the player on

our site. The next time your player goes to check and see, it will find something new there and it will ask you if you'd like to get the newest update.

Bob: Does it have to go to the Apple site to check and see, or is it doing this automatically?

Frank: The checking is automatic, it's transparent, you won't even know it's happening. We want everyone to have the latest possible software that we have. If we have it on our servers, you deserve to have it on your computer and we want to give you that option. So, every other week or so, it will go back and check for you. We aren't going to go and change it every day, because that would be revision control madness. It may happen every month, every couple of months, when it is worth signalling people that there is an update. If we find something nasty, and someone on the field reports it and we're going, 'Son of a gun, we missed that one,' we can usually fix these bugs in a day, once we find them. We can then post that to our servers, so that people can be signalled, once they go online. So it's a good mechanism for us to get in touch with people. It's not a marketing tool, it's nothing more than an installer update.

Bob: So I launch my PPP and it's going to happen then? Or it's going to happen when I launch my browser?

Frank: Once your system knows you have an Internet connection.

Bob: If I have a connection and I go to any other site, part of it will go to the Apple site?

Frank: Yes, if QuickTime is running, and you are actually looking at some streaming media. It's not just when you are surfing around, it's when you are surfing and looking at some streaming



FRANK CASANOVA:

A Conversation about Apple's QuickTime 4

CONTINUED FROM PAGE 13

media content. It's not going to be in some hibernation state and sniff the Internet and jump to life. It's going to be up and awake and aware of what's going on and it's going to dispatch that command.

Bob: You would have to be at a site using streaming server software?

Frank: All you have to do is click on BBC World News or other news. Just open your player and click on one of these channels. People do this constantly, and as soon as you do that, the installer will go back and check. It's not a sophisticated system where you have to figure out where you have to go. You just click and there you go.

Bob: Give us a step-by-step description of how you would compress video using a codec.

Frank: Take, for example, the video you are taking of me right now. Presumably, you would take this video and import this into a QuickTime-aware application such as Premier, Final Cut Pro, or one of the quality non-linear digital video editors. Once you've captured it, you bring it in as a QuickTime file, you edit it, you move things around, you put special effects and transitions and swipes, then you actually prototype it that way, and then you save the file. When you have the finished file, you would hint it. Hinting is the process by which you make that file streamable. So you can take that file, put it on someone's hard disk and you can http download it because there's nothing special about that, it's just a QuickTime movie. Now you want to stream it into real-time, so you would hint it using a variety of tools. If you upgrade to QuickTime Pro, for \$30 (US), you could export it as a hinted movie and it adds a special track to the movie that's invisible unless you are doing real-time streaming.

Bob: How would you know it's been hinted?

Frank: You could go to Get Info and look at the file type. It will say audio track, video track and also hinted audio, hinted video. You know that those two extra tracks are invisible unless

it finds a real-time streaming server, like a QuickTime streaming server, and those two tracks will jump to life and parse out the movie. So hinting is nothing more than resaving the movie.

Bob: In one file, I will have one hinted track and one regular one?

Frank: It's not a whole other file. Basically, it sees the DNA of the video track. It understands when the major keyframes are, it looks at it and analyzes the actual video and audio tracks and breaks them up into pieces, such that QuickTime knows to pump out the audio and video to the client in a synchronized fashion, regardless of where it's going to work or where it's coming from. Of course you have to compress it at some time, using Media Cleaner Pro or one of the compressing tools that use Sorenson video or Q-Design audio or Qualcomm for speech. Compressing music and conversation are very different. The conversation we're having now would be very easy to compress. There are a lot of pauses and my voice isn't going through three different octaves. Music is very difficult to compress and that is where Q-Design and Sorenson (which are exclusive to Apple) come into play. They are the best codecs that we have seen in the industry.

Bob: If we were to build a site in a studio and the person that was creating the video wanted to have his own stuff on his own server, on his site, because he was constantly updating it or doing something live, what equipment and software would he need to purchase, above and beyond a video machine, a computer and the camera?

Frank: There is very little other equipment that you need beyond that. It depends on whether you are doing video on demand or live broadcasting. QuickTime 4 supports both formats. Live with the camera straight to the Web, where everything gets compressed, or video on demand, where content is stored on a hard disk and served up as people want to see it. For video on demand, we would go through all the steps we talked about before. Typically, we would want to have high quality microphones, and compressor hardware that gets rid of the spikes and compresses the sound into a fine range. You might also want to have a mixing board that controls the levels of the audio that's going out. So it's very straightforward. Recording voice is simpler, because there are fewer vagaries, like less spikes. So there's minimal investment to do the audio. For the video, it's just a camera going right into the

computer.

Bob: What equipment do we need to build a streaming server?

Frank: As with anything in this industry, you get what you pay for. If you invest the time, energy and money into professional editing equipment, you will get much better results. If you want to take shortcuts, you will get lesser results, but nonetheless, it still works. What I mean is that in the area of audio and video capture, you will want to capture the audio and video separately. You might want to use high quality microphones that could capture a wide range of audio and more subtleties, as opposed to a microphone on top of a camera. The better the source material, the better the results will be at the end of the line. Once you have the audio and video captured, and the levels are adjusted properly, and you used a non-linear software like Final Cut Pro to mix and match, do special effects, etc., and you have a file at the end that you are happy with, you would use Media Cleaner Pro to compress the audio, compress the video to create the hint tracks that tell the QuickTime streaming servers how to parse out that file. Once you have it on the QuickTime streaming server, it streams—it just streams. I don't go through all the pain and agony that a professional goes through to get the best quality because I'm a hacker when it comes to video producing. What I'll do is just take some raw video, hint it using the QuickTime player, which is less professional than using Media Cleaner Pro, which has a lot of settings. The QuickTime player has a variety of settings, but it has a best case scenario set, where I just save as a hinted movie, get the output file, FTP it to a server and I'm off and running. For me that's good enough, but if I'm a professional, I will want to go the extra step with the professional tools. We ship Sorenson Video and Q-Design audio as a part of QuickTime, but Sorenson has a pro version of their video codec, and Q-Design one of their audio codec, both of which add far more performance and capability that a professional would want to use in preparing their media.

Bob: Is hinting done by QuickTime or by Media Cleaner Pro?

Frank: Either of the applications can actually apply the hint track.

Bob: If you get QuickTime 4, that will do the hinting, but what if you've updated Media Cleaner Pro to a later version?

Frank: Media Cleaner Pro announced a new version of their product as we introduced

QuickTime 4. They had to wait for us, because they use QuickTime as part of their product. They may actually use our hinter. The process of hinting is an intellectual property that Apple has and that we have included as part of the product. After the file has been hinted, that same file can be put on a CD-ROM, DVD, a hard disk—it doesn't matter, because when the hint tracks aren't seen by a QuickTime streaming server, they are just ignored. It's really cool, because we don't have to convert or transcode any of our content, any of our files to any special format, the way Microsoft goes from AVI to ASF, and it creates this non-editable piece of media, that they can't do anything with, other than modify the source. We don't do that. Our one QuickTime file, from capture through to archive, is a non-destructive format. The same thing applies across the entire life cycle of a piece of video and audio.

Bob: So, how does hinting actually work? What's going on in the background?

Frank: Hinting is, in its simplest form, understanding the DNA of the movie. It parses up, depending on the parameters you set. Within the QuickTime player, there's a variety of settings for frame rates, size of movie and target deployment. The hinter will look at the settings you've set up and it will break up the movie in the way that you have specified, so that the file works best for the deployment that you want to use. T1, 28.8, these are very different settings, they have very different frame rates. So, if you know you are going to do a music video at 28.8, you might only want to do one frame per second on the video, and set that up. You might want to give a little more bend to the audio, if it's a music video. You might give up a little video quality, in favour of audio. You can set that all up and hint it, and you will get a file that matches all of your settings. You can play it, and if you don't like it, you can go back and do it all again. You treat the settings until you get that sweet spot, and the hinter takes advantage of all of that and it parses the movie out. When it finds itself on a QuickTime streaming server, it broadcasts its way out to your customers in the way that you intended them to see it.

Bob: So this is a lot different than setting your QuickTime settings for a cable modem or 56K. You've told QuickTime what it is, and it goes and selects the appropriate movie for the playback. It's telling the server which one to give me, so this hinting is different.

Frank: Hinting is done in every streaming movie. What we have done that is different is

that we have allowed the QuickTime player in its Pro version to allow the user to set up what bit rate they want. So, for example, if you go to a lot of websites today, you'll see a lot of videos' servers up at 20KB, 40 KB and 80KB, that means 28.8K, 56K and ISDN. We use that nomenclature right in the QuickTime player, so if you know you're targeting ISDN, you can use 80KB. Even though ISDN is faster, the effective throughput is considerably lower, closer to 80KB, so we encode the content at 80K bits per second and the QuickTime player does all that for you. If you take one piece of video, you'll encode it three different times—20KB, 40KB and 80KB—and you create one reference movie. That reference movie is basically the front to the three other movies. When I'm sitting at the QuickTime player or my Web page using the QuickTime plug-in and I see a piece of content that I'm interested in, I click on that content and my client and the server will have a quick discussion and it will say, 'What settings do you have? How fast are you going? I'm going at this speed and I have these movies.' Great! I'll take the 20KB one, cause that's what I'm running. Transparent to the user, you will get the movie that is best for your connection speed, because the QuickTimes are talking. The same is true of language. If you author a piece of content in multiple languages, French, Spanish, Italian and English, and as a client, your operating system is set to French, when you hit that piece of media the QuickTime will look to see which OS is set and it will give you the media, set to the audio track, set to whatever language your OS is. Only in QuickTime can this stuff happen. It's sophisticated, it's elegant and it just works. It gives you the best experience you can imagine.

Bob: I currently go to the BBC site, and I have a cable modem that's pretty quick, but I'm getting a very low frame rate and I know that I can get video over the Internet much faster than that by just going to another QuickTime sample site. Why is it like that?

Frank: BBC is a live feed. What we have is a company in New York City that is taking a satellite downlink, a live feed from BBC World News, that is unlike video on demand. That stream comes down to a satellite and receiver to a building in New York City, down through their cable, and it goes into three different servers. Each of these servers is running what's called a broadcast tool; it was written by Sorenson, who also do our video codec. The broadcast tool is running at three different

settings, 28.8K at 20KB, 56K at 40KB, and ISDN at 80KB. Maybe we'll run it a little hotter than 80 KB, maybe a little closer to 90KB. So you come in with your cable modem, and you click on BBC news; we don't have a cable modem speed or stream for you, because you know what? Most people don't have cable modem, most people are still at 56K, in the office they're at ISDN, some people have T1, but we're not even putting them out at T1 speeds because not enough people will look at it, and it takes time and money to set up yet another server to serve yet another bit rate. So we have to pick at the curve: 28.8K at one end, ISDN at the other and most of the world is at 56K...Anybody can author at any speed, we don't care. If you, for example, want to provide only cable modem encoding, go for it, it's not a problem, but only a few people will be able to look at it. Maybe you want to only provide for DSL and cable modems—not a problem, if you do everything at 1.5 megabit and up, you are going to get an incredible experience. QuickTime will scale beautifully up to cable modems. The problem is that the number of people who can actually view your content will be somewhat limited. If you have a clicker, a club or a special event over a closed network, then go for it and you'll be incredibly happy with your results, but if you want to get your content out to the broadest distribution, you have to acknowledge that 28.8K and 56K are the predominant connection speeds on the Internet today.

Bob: So, the easiest way to cover all bases would be to have several servers.

Frank: Correct.

Bob: But to get this great quality Sorenson has been rendering all night long. How do we getting around this?

Frank: Well, you give up some quality. Live broadcasts tend to have less quality than the video on demand broadcast, so you're right—it's on the fly, it's live, it's on the software, so you basically have to get as much as you can encoded, as fast as you can. The faster your hardware, the better your performance is going to be. There are some cards you can add. There's a company called Ice down in Massachusetts that has a card that you plug into the Macintosh. When you run Sorenson, the Ice card has eight CPUs on it that are running the Sorenson codecs. So the frame rate goes way up, because all these CPUs are munching on the data and spitting out more frames per unit

CONTINUED

of time than we could do in software only. So it's all a matter of how much performance, how much time. Broadcasting is real-time, so you don't have a lot of time cause it's gotta happen in a heartbeat. The more times you can go over that information, with more CPUs, like the Ice guys with Sorenson—we can get a higher frame rate, which will appear to be a smooth experience of the video.

Bob: Will the Ice card run in real-time?

Frank: Well, everything is real-time, it's only a matter of how much information you get in that real-time. So one frame per second is real-time. The experience is somewhat lacking, but it's real-time, nonetheless. Twenty-four frames per second is what you get when you go to the movie theater, and it's real-time. The experience is incredibly dramatic, it's deep, it's rich, the colors are there and the lip synch is dead on. Those are the two ends of the extreme and everything in between is based on the performance of the computer, your encoders, and your connection. So it's all performance based.

Bob: So, for example, if I want to send this video out to a cable modem, and I want it to look as good as possible, is the Ice card going to help me there? Is it going to render live?

Frank: The Ice card is good for live broadcasts. If what we're going to do here is encode for video on demand, then the Sorenson codec will munch away at the video in non-real-time, it will take longer than real-time to create high quality video and you'll do it in multiple bit rates, such that viewers will connect at whatever speed they have. So the Ice card doesn't really add any value there, it adds value when you are trying to capture live information. It's meant for live broadcasts where more CPUs are great.

Bob: How many cards? More than one?

Frank: You know, we actually haven't tried using more than one. The CPUs on the Ice cards keep getting faster, right now there's an array of eight of them. That's a good question, I haven't actually asked them whether multiplying those cards would make it even better.

Bob: So an ideal situation would be to have a good quality camera, locked off so there isn't any background motion, so it's a talking head situation like learning on demand?

Frank: Correct.

Bob: Cut into and switched into bar charts, something like that?

Frank: Correct.

Bob: Then compressed into something you have, say a G3 computer that doesn't neces-

sarily have to be a server but it's a computer doing all the work involved in compressing, with at least one of these Ice cards helping the computer along...

Frank: For the live broadcast...

Bob: ...and now I have to deliver it to a thousand people, so I don't want the computer that's been doing the compression to be serving at the same time.

Frank: Exactly.

Bob: What do we need to do next?

Frank: So, we've done everything we've talked about thusfar. The image is being captured by a camera, the cable goes into the back of a computer, or multiple computers at multiple bit rates, and the Sorenson codec, with the aid of an Ice card perhaps, is compressing the audio and the video. At the other end of the computer, it's being spit out onto the Internet. It goes first to a server that reflects the stream. So the Sorenson broadcast tool generates what's called a multi-broadcast stream. This is a stream that is effective utilization of bandwidth and it goes out to whatever part of the Internet is multicast enabled, and it will propagate as far out to the end user as it will possibly go, until it hits a router or it gets further out to the Internet where the streams actually turn into unicast streams, which is a personalized stream. The QuickTime streaming server with the Sorenson broadcast tool can broadcast a multicast stream, which is effective utilization of bandwidth, and the QuickTime streaming server can actually be a reflector, which takes that multicast stream and knows that it has reached the end of the multicast line and then splits it out into respective unicast streams—one for you, one for me, one for anybody that wants to watch it. The cool thing about the QuickTime streaming server is that it does both the serving and the splitting and it does video on demand. The server is multi-purposed, multifunctional and it's free.

Bob: If you had the ultimate way to deliver this, and you were going to buy gear and put in an installation of this type of technology at, say, CNN—what would the ideal be? Going from CNN from a T1 line to a fibre-optic backbone, have a server there?

Frank: Remember, CNN is caught on an anomaly in this game because they have more traffic than pretty much anyone else for live and for video on demand news, so they will opt to use high performance workstation servers, such as Sun or IBM, all of which stream QuickTime, so we're happy about that. A smaller installation could use a blue and white

G3, the standard machine we sell to desktop publishing and engineering people all day long. They can use that as a streaming server, there's no problem there. They would prepare the materials as we discussed, put them on high performance servers, and put a live link into their Web page. You would be reading the text and it says 'Watch this activity'. You would click on it, either right in the Web page or in QuickTime player itself, and the video would just jump to life. Either in-line in a browser, or in a window they've reserved in their page, or the QuickTime player would come up and they would be able to watch it, assuming you have QuickTime on your client side; because when you install it, you get the QuickTime Player as well as the browser plug-ins that enable all of your browsers—Netscape, Explorer and AOL—to be QuickTime streaming containers as well.

Bob: Is there anything important that we missed?

Frank: Actually, we've covered a lot of good authoring information. QuickTime is fast becoming incredibly ubiquitous in the streaming media world. What is important to understand is that upwards of ninety per cent of Real Networks' streamed video starts off as QuickTime. Many of your readers may author Real content, and they know that they capture and edit and archive in QuickTime—but until QuickTime 4, they couldn't deploy in QuickTime streaming. Now you can, now you have the complete workflow—ready, willing and able to capture and archive and everything in between, so you're going to see a huge number of QuickTime content authors using QuickTime to do streaming. Number two, the streaming of QuickTime is free. Unlike Real, who charge you a server tax to stream the content, our QuickTime streaming is free. We've actually open sourced our QuickTime streaming server—you can go to our website today and download the source code for free, and compile it into whatever flavour you want. And if you buy the QuickTime streaming server from Apple, which is part of Mac OS X, it's \$500 (U.S)—it's nothing, and that one server, on a blue and white G3, can serve up to over 1000 modem rate connections at a time. So we have the architecture that goes client to server and everything in between; we're building some really cool content with our partners; and we're serving in a way that's never been done before, totally for free. So, QuickTime—it's new, and it just totally rocks, and we invite you all to download it! *