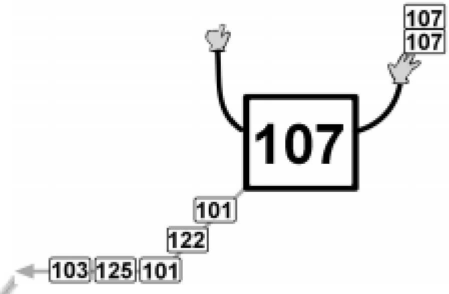
Chapter 6

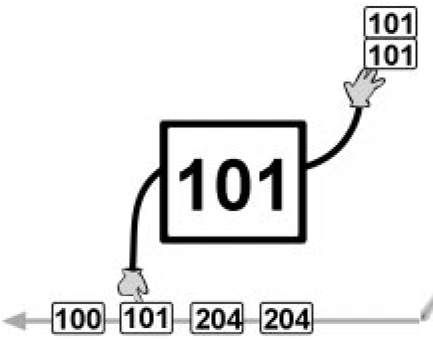
WHAT IS STREAMING MEDIA AND HOW DOES IT WORK?

RealPlayer 8 Plus and RealSystem® G2 with Flash are a huge leap ahead for streaming technology on the Internet. RealSystem G2 is the collective name for the newest generation of RealNetworks' collection of servers and tools that create and serve content to your RealPlayer. Of course, most of the bits and pieces of RealSystem G2 are hidden from you when you watch or listen to clips. What follows is an explanation of how streaming media works and how RealSystem G2 and RealPlayer fit into the scheme of it all (a little techy but in an easy-to-understand way; we want you to be as excited by what's going on as we are).

Networking



To understand how RealPlayer works, you first need to understand *networking*. In its most simple form, networking is one computer exchanging information with another machine. This isn't much different from mailing a letter.



The first computer addresses a piece of information to another computer and then sends it across the *wire* (the cable that connects computers to one another). The second computer checks the addresses of all of the messages coming by and reads only the messages addressed to it.

Of course, when you have many computers and a great deal of information this process gets more complex. The essentials described above, however, are the same no matter the size of the network. Every computer has a unique address and all the *bits* (pieces of information) have an address to which they will eventually be delivered. (For those that know about multicast, that is discussed in Multicast on page 53. For those that don't, be patient, we'll get there.)

A piece of information can be very small, such as a simple 'yes' answer, or very large, such as your favorite song. If everyone on a network were given exclusive use of the wire until the whole message was sent or received, it wouldn't work very well. Can you imagine trying to get your e-mail and having to wait while someone else listened to Beethoven's Fifth symphony? To avoid this problem, messages are broken into lots of little pieces of information that are called *packets*. On most common networks each packet is the same size and is addressed as described above.

This process may sound like extra work, but it keeps the flow of information smooth and constant with fewer bottlenecks. It is easier for a number of same-sized cars to merge onto the freeway from the same ramp than it is for a bunch of cars and an 18-wheel trailer to merge on from the same ramp. In the latter scenario, a large number of cars will have to wait as the trailer passes and then wait again while the trailer waits for room on the highway so it can finish its merge.

Handling the flow of information are a number of specialized traffic cops: routers, brouters, bridges, servers, gateways, etc. However, there are only two that you need concern yourself with: servers and routers.

Servers are computers that store information (e.g., videos, e-mail) and, in a company setting, programs (e.g., word processors). These computers *serve* the information and programs to other computers on the network to which they're connected. Every network has at least one server and sometimes several servers.

Routers scan the flow of information, read the address on each packet of information and sends each packet along the appropriate **route** (thus their name) to get to its final destination.

Internet

The Internet is not a single specific entity. You can't point to one thing and say, "That's the Internet." The Internet is an interconnected collection of networks that speak to each other just as a network is an interconnected collection of servers and computers. Sometimes this connection is physical, such as a wire or fiber-optic connection, but sometimes the connection is via satellite or microwave. This might sound familiar. When you pick up your phone to call someone, the signal sometimes travels by wire to the person you are speaking to, but it is often sent via microwave tower or satellites, depending on how far away and where you are calling.

Some (very) quick history

[Yes. You can skip this, but didn't you always wonder?]

Unlike the phone system, no one set out to create the Internet as we know it today. In fact, it didn't even begin life as "the Internet", but rather as *ARPANET* (Advanced Research Projects Agency Network) in 1969.

Note

To be wholly honest, no one set out to create the American phone system as it is today either. It was a number of smaller, private companies that eventually merged into one giant company... before it became several smaller companies again.

That was then

ARPANET's original purpose was to help universities, the American military and the U.S. Defense Department contractors maintain internal communication (even during natural disasters and wars) and exchange information easily. ARPANET began life with four servers, a far cry from the eight million or more servers that comprise the Internet today.

In 1973 ARPANET became DARPANET (**Defense** Advanced Research Projects Agency Network). DARPA pioneered the research of how to best exchange information across this "internetwork." Their research led to the development of *TCP/IP (Transmission Control Protocol/Internet Protocol)* that, for the purpose of this discussion, is currently the standard way all those packets of information your computer sends out are addressed so they get to the right place all together and in the correct order.

Universities were quick to exploit this amazing infrastructure of communication. Around the core of the secure DARPA sub-system, Universities began adding additional servers and connections; the *Internet* was born. For many years the Internet was not much more than a massive e-mail exchange and bulletin-board system (posted messages people could respond to in kind).

This is now

In 1990 CERN—European Laboratory for Particle Physics, with the help of Tim Berners-Lee and the ideas of Ted Nelson, began prototyping a standardization for information sharing that became known as the World Wide Web.

Additional Information

You may recognize that the "www." in most common Internet addresses is in fact World Wide Web. To this day Berners-Lee still shapes and directs how the Web develops through the W3C (World Wide Web Consortium, **http://www.w3c.org/**).

While the Internet is made up of physical entities, the WWW is more an agreed upon collection of *standards* (or *rules*) about how to address packets and how to create Web pages with *Hypertext Mark-up Language (HTML)*. Before these standards, everyone had different formatting codes, much as different word processors create files that other processors may or may not be able to read. HTML gave everyone a way to share information easily. Information began to flow at an increased rate.

Additional Information

You may have noticed that every Internet address begins with http://. HTTP (Hyper Text Transfer Protocol) is another kind of standard, or *protocol*. HTTP tells your browser and computer that HTML has been sent to it so it can read the incoming information. HTTP gives the server and your computer a common dictionary so that the information coming in makes sense.

In the case of your RealPlayer you may notice that the addresses start `PNM://` (Progressive Networks Media) or `RTSP://` (Real-Time Streaming Protocol) which your RealPlayer recognizes and speaks. PNM is an older protocol but many clips are still around in that form. Any newer, RealSystem G2, content will have `RTSP://` before the address of the clip. Edit a Favorite to see its address (URL). These protocols become important if you are using your RealPlayer at work because of firewalls (see Firewalls and Security on page 54).

Equally as important, cheaper, faster hardware became available in the mid-80's and early-90's to the public at large. The Internet exploded from a text-based system to one that contained pictures, sounds, videos and animations. Why? Simple text (words and letters) requires very little information to reproduce. It is easy to send around whole pages of information in relatively few packets quickly, even at slower connection speeds. Sound files can be small, but it depends on their quality and length. The same is true for pictures and small animations. Large animations and video have a unique problem. Both large animations and video are essentially a whole bunch of still pictures that need to be seen at a very quick rate to give the illusion of movement.

Until recently the best way to ensure that video and large animations worked well over the Internet was to download the whole file and run it from your machine at home. If you downloaded the file yourself, the download time was obvious; you had to wait a long time till the file had finished saving. Sometimes, though, you would just be waiting a **very** long time for the movie or large animation to begin. During that time it was being downloaded to a temporary file on your hardrive before it started to play.

Streaming

Enter *Streaming*. Streaming takes files—whether they are sound, video, animation, or other media types—breaks them up into smaller pieces and sends them to their destination. This is very similar to how computers send information across a network or the Internet in general.

You may be scratching your head now asking, "But you just told me that information is sent to my computer in little pieces anyway. What's the difference?" A good question.

Figure 6-1: Sending a book page by page

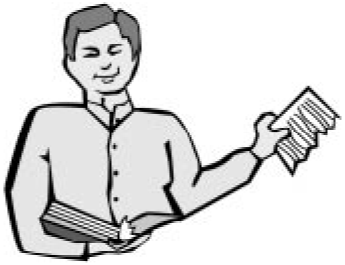


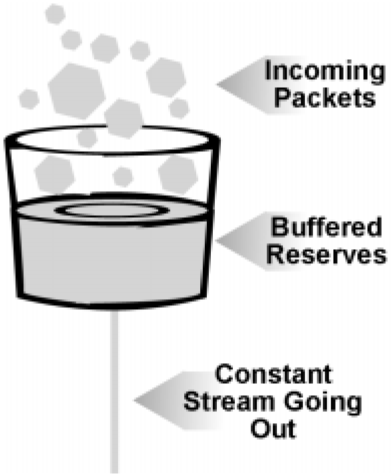
Figure 6-2: Receiving a book page by page

The answer is what makes your RealPlayer so exciting. RealPlayer is able to read the file stream **as it is coming in** and begin playing it long before the rest of the file arrives. Imagine reading a novel as someone hands you page after page, rather than waiting for them to finish the whole book and handing you the entire novel at once.

Of course, if the file was simply read and played for you as it came in, you would have a good number of interruptions; think about how often you have to wait for a Web page to display on your screen. Now, consider that Web pages are much smaller than most media files (audio, video, animation, etc.). RealPlayer combines another technology with streaming to make playback smooth: *buffering*.

During buffering a whole bunch of packets are collected before being played.

Figure 6-3: Buffering



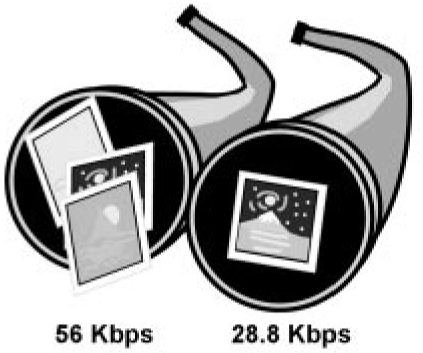
Imagine a cup being filled from above in fits and starts. A small hole in the bottom of the cup allows a constant stream of water to drain from the cup. As long as there is enough water backed-up in the cup, the stream will continue to flow out at a steady rate (see Figure 6-3: Buffering).

As RealPlayer begins to play the file, it continues to collect packets in reserve. This means that even if there are minor delays in getting the information packets to your computer, your experience of the music will be continuous rather than having it stutter along erratically.

So what is a stream?

Above we explain what streaming **does**, but not exactly what it **is**. At least, not what it is for RealPlayer. Each *stream* carries specific types of information. When you watch a video the visual part, what you watch, comes to you in one stream while the sound comes in another. RealPlayer synchronizes these two separate streams so that the right words come out of a newscaster's mouth when they should or so that music crescendos appropriately when the action requires it.

Figure 6-4: Bandwidth



Additionally, streams can be optimized for different *bandwidths*. Bandwidth is, essentially, the amount of information that can pass through a particular point of the wire in a specific amount of time. The higher the bandwidth, the greater the amount of information that can come through (see Figure 6-4: Bandwidth). The speed of your modem, if that is what you use to connect to the Internet, determines the bandwidth of the stream that you can receive.

A 28.8kbps modem can receive, approximately, 28.8 thousand bits per second. A 56K modem is capable of receiving, approximately, 56 thousand bits per second—almost double the capability of a 28.8 modem (remember, a *bit* is a single binary number). Of course these capabilities assume that your phone connections are perfect, which few are. In other words, your actual mileage may vary.

You may, incorrectly, hear people refer to these speeds as the *BAUD-rate*. However, BAUD is actually how frequently the base sound (usually called a *carrier wave*) changes per second. Information is interpreted by how that sound changes. Originally, a single change, say from a high tone to a low tone, was a single bit of information. Because of improvements in compression, many more than just a single bit of information can be transmitted per change in the carrier frequency. The important point is that the higher the bandwidth you can receive, the higher the quality the sound or video will be.

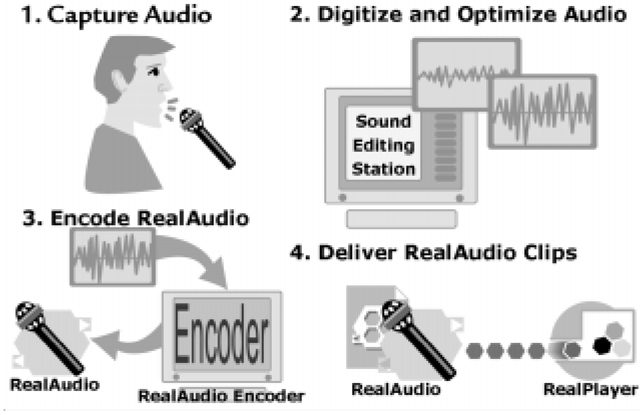
Additional Information

BAUD is named for French inventor Jean-Maurice Emille Baudot (1845-1903). Among his other accomplishments, Baudot patented in 1874 the telegraph code that eventually supplanted Morse code (Morse code fell out of general use by 1920). Baudot's new code was a 5-unit combination of on/off signals of equal duration as opposed to Morse's long/short, dash/dot. The code consisted of 32 five-unit characters covering each letter of the alphabet, punctuation and additional machine controls. The modern version uses 7 or 8 on/off signals or "bits" that have 128 characters and is otherwise known as ASCII code now. Not content with simply improving transmission speed and accuracy, in 1894 Baudot invented a distribution system for simultaneous (multiplex) transmission of several telegraph messages on the same circuit, which is now common practice for all internetwork communication.

Where do streams come from?

Streams have to be created by someone for you to watch them with your RealPlayer. This means that an original performance or broadcast was recorded and then turned into a *stream*. But how exactly does this happen? Looking at Figure 6-5: Creating an audio stream, you can see the four steps necessary for creating any kind of content.

Figure 6-5: Creating an audio stream



Step 1

The performance is *captured*. Captured simply means *recorded* or *created* in some way so that it can be changed later into a stream. With some content (e.g. animation) there is no performance, only the artist's drawings which are created. The idea, however, is the same. The originals are captured on the computer so they can be changed at a later time into a stream.

Step 2

The captured performance is edited so that the presentation you receive is what the content provider wanted to show you.

Step 3

The edited performance is *encoded* (see Encoding below) into a stream. Essentially this means that the file is prepared to be streamed.

Step 4

The encoded file is placed on a RealServer (see How does streaming occur? on page 49) so your RealPlayer can play it for you.

Additional Information

This holds true for live broadcasts as well. As with live television, the performance or event is viewed by the cameras and/or microphones and is converted as it happens to a form that can be sent to you a few seconds later.

Encoding

The process of getting a file ready to be streamed is called *encoding*. Encoding takes a file and breaks it up into readable packets so it can be sent and read "on the fly" (remember Figure 6-1: Sending a book page by page on page 43?). Every clip, or every stream in a clip, is encoded for a specific bitrate (see also SureStream on page 51). The higher the resolution (the more information) a stream has, the higher the bandwidth (the larger the hose) it needs to get the information through to you at a useable speed.

Additional Information

Encoding is performed by special tools such as **RealProducer®** by RealNetworks available at **http://www.realnetworks.com**RealPlayer is able to read these files and encode new files when it is recording.

When you are playing a clip, you can see at what bandwidth a stream has been encoded by looking at your Status bar (see Figure 3-11: Status bar on page 15). The number shown is the stream bandwidth you are receiving at that moment. For a single audio stream this is straight-forward: if you are receiving a 20Kbps (kilobits per second) audio stream, that's what will show up on your

Status bar. What if you are receiving multiple streams? The answer is: Stream bandwidth is additive. For example, if you are watching a Macromedia Flash animation encoded at 12Kbps that also has an audio stream encoded at 10Kbps, the bandwidth will appear on the Status bar as 22Kbps (see Figure 6-7: Multiple Streams on page 51).

The *pipeline*, or bandwidth available, from your modem has a set limit, you can't make your modem work faster than it is capable of working. Content providers usually encode their streams to handle a variety of bandwidths. The higher the bandwidth, the more information, the better your experience. RealPlayer uses buffering (RealPlayer Plus can also use PerfectPlay, see PerfectPlay on page 37) to overcome minor slowdowns on the Internet. But to watch a stream encoded as a 56K stream using a 28.8K modem you would have to buffer a large amount of information. Buffering means waiting and using a good deal of memory.

Codecs

If something is *encoded*, you must *decode* it in order to read it. To decode information on your computer you use a *codec*. Codec, like 'modem', is a word created by combining two actions of a process.

- **Modem**: modulator/demodulator
- **Codec**: coder/decoder

Modems convert digital information from your computer into analog information (sound waves) that can be sent over traditional phone lines. Codecs read digital information and process it into analog information for your player so you can listen to it. CD players use codecs that convert the digital information on a CD into sound waves that you hear on your speakers or headphones.

Your RealPlayer decodes each specific type of file (video, text, etc.) with a specific codec, or *plug-in*.

Additional Information

> You can see what plug-ins and add-ins you have by looking on the **Upgrade** tab of your **Preferences...** (see Upgrade Preferences on page 63).

When you first install RealPlayer you will have several plug-ins: RealAudio, RealVideo, RealPix, RealText, and RealG2 with Flash. Each of these different types of media is encoded separately and sent in separate streams that your RealPlayer can decode, synchronize, and play.

From time to time we will update one or more of these plug-ins or even add new plug-ins to enhance your RealPlayer.

Tip

The latest plug-ins are always available via **AutoUpdate**. Just select **Check for Update...** from your **Help** menu. Also, if you attempt to play a clip that requires an updated plug-in, you will be notified and offered the opportunity to upgrade.

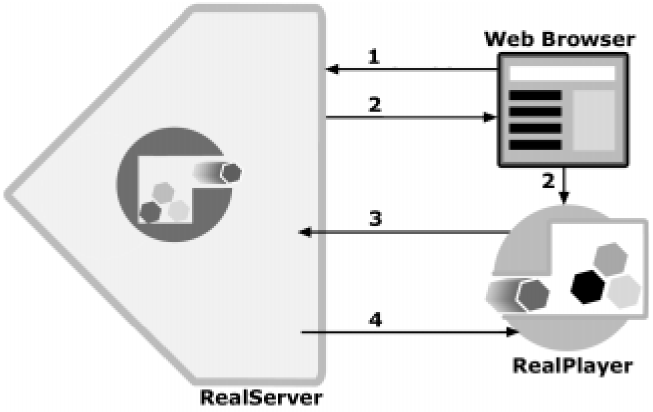
How does streaming occur?

Ok, we now have streams of data being encoded and decoded by the plug-ins and RealPlayer turning all of that information into rich presentations— but where do these streams come from and how does your RealPlayer find them?

Remember, the Internet is a simply a bunch of connected *servers*. Servers *serve* information to other computers (this is a loose definition but accurate for the moment). A *clip* is any number of streams that make up a presentation. As we discussed earlier, a clip can be made up of more than one stream, such as a video stream and a sound stream. Clips themselves can be a single entity or several clips in a row. This is why we say, *clip* or in the case of multiple presentations, *multi-clip*.

So what happens when you click on a link in a web page to hear a clip or multi-clip?

Figure 6-6: How RealPlayer connects to a RealServer



Step 1

All Web pages are stored on special servers called web servers. When you click on a link in a Web page to listen to a clip (whether live or pre-recorded) the web server sends your browser a small pointer file that is associated with that link.

Step 2

Your browser recognizes that the file isn't just another page so it checks to see what program can read it. When it recognizes that it is a RealPlayer file, it starts up RealPlayer and passes the file off to the your program.

Steps 3 and 4

The file is very small and contains only the name and location of the clip on a *RealServer*. Your RealPlayer contacts the RealServer which sends the actual stream and stays in contact with RealServer so you can fast forward or rewind or otherwise navigate through the clip. In the case of Channels and Live Stations, RealPlayer speaks directly to the RealServer without the need for the earlier pointer file.

RealServers are similar to Web servers but rather than serving pages to browsers, they serve streams to RealPlayers. RealServers work in concert with Web servers to bring multi-media to networked environments, such as the Internet.

Note

> Why "networked environments" rather than Web or Internet? Because RealServers™ are also being used on company networks to make company-wide addresses, training, and other information easily available to employees. RealServers are available for download at **http://www.realnetworks.com**

RealPlayer and RealServers can communicate with each other so that RealServers can pick the appropriate streams to send you. If a RealServer attempted to send a stream that was encoded at a high bitrate (for example T1 [a really fast connection]) and you only had a 28.8K modem, you would not have a very good experience.

SureStream

So how do you get the appropriate stream for your RealPlayer when you try to play a clip? RealPlayer, as mentioned above, can talk to the RealServer that is sending the stream. In a more exact view of the process:

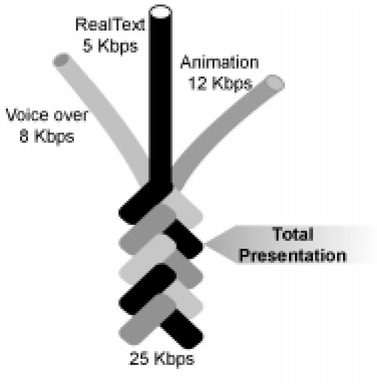
1. RealPlayer sends the **Normal** and **Maximum** bandwidth settings from the **Connections** tab of your **Preferences...** to the RealServer serving the clip.
2. The RealServer automatically selects the streams encoded at a bitrate closest to your **Normal** setting without exceeding it. If your actual connection is faster than **Normal**, the server will upshift to a higher bitrate if it is available.

This ability to upshift is a RealSystem G2 feature called *SureStream*.

Prior to RealSystem G2, whatever bitrate you connected to initially was what you were stuck with for the entire clip, even if your connection got faster or slower. Because streams in the new RealSystem G2 can be encoded for several different bitrates within the same file, RealServer is able to intelligently pick the best possible stream for your current connection.

SureStream also monitors your connection for you. If it senses too much congestion, it will downshift certain streams. Content providers can choose which streams they feel are more important to your experience (for example voice quality over animation). If SureStream senses that your connection is better than expected, it will upshift to a higher bandwidth on all or some of the streams to bring you a higher quality presentation. Upshifts and downshifts will happen automatically.

Figure 6-7: Multiple Streams



SureStream becomes especially important when you are viewing some of the newer presentations that RealPlayer is capable of playing. When playing a SMIL presentation (see SMIL below) each of the different streams for each of the components of the presentation has a bitrate associated with it. The total of all of those bitrates is the lowest connection speed you need to have to play the clip without excessive buffering. For instance, a presentation that had an 8Kbps voice-over, a 12Kbps Macromedia

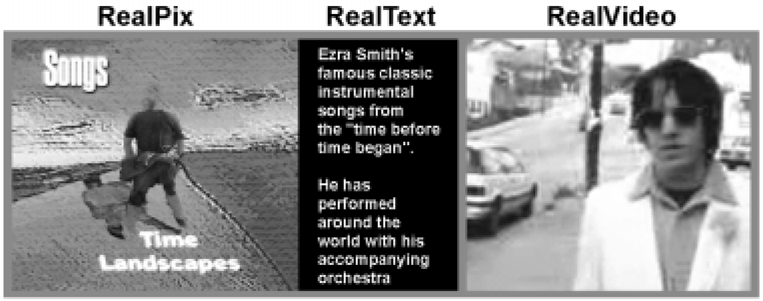
Flash animation component, and a 5Kbps RealText component would need at least a 25Kbps connection (8+12+5=25) to play without buffering constantly. Each stream flows into the same "river" and adds to the whole presentation that your RealPlayer plays (see Figure 6-7: Multiple Streams).

Of course, RealPlayer buffers for you automatically to help you play higher bitrate streams. So even over slower connections you can experience superior quality media. The price you pay to watch higher bitrates over slower connections is the time it takes to buffer, which is still much shorter than it would be to wait for the whole file to download.

SMIL

Now the fun begins. RealPlayer can play multiple clips presented simultaneously as a single presentation. This is done using SMIL (pronounced "smile," Synchronized Multimedia Integration Language—you see why we stick with acronyms?). SMIL is one of the newest language sets on the Web. While HTML (Hyper Text Mark-up Language) controls how Web pages present text and pictures on your computer screen, SMIL controls how *complex* media presentations should be streamed to and laid out within your RealPlayer.

Figure 6-8: SMIL presentation



Ezra Smith's famous classic instrumental songs from the "time before time began".

He has performed around the world with his accompanying orchestra

By "complex" we mean a presentation that includes more than one clip being received by RealPlayer which is then synchronized with each clip being presented in a separate space. For instance, you could watch a music video (RealVideo), see the lyrics (RealText), hear the music (RealAudio), and browse stills from the concert (RealPix) all presented to you at the same time (see Figure 6-8: SMIL presentation).

In fact, the Tutorials that introduce the new RealPlayer to you are SMIL presentations that include RealAudio, RealPix, and RealText. Each element of the presentation is created and *encoded* (see Encoding on page 47) separately and then the lay-out is described to your RealPlayer using SMIL so it can present them together.

Additional Information

To learn more about SMIL go to:
http://www.w3c.org/AudioVideo/

Multicast

Multicast is the organic chemistry of streaming. If you had the joy of taking organic chemistry, or almost any higher level science class, the first thing you are told is: Forget everything you've been taught before; because it's wrong.

Well, we won't be quite that cruel here, but *multicasting* is a unique process in the network environment. In networking, every packet of information is uniquely addressed to a specific computer. Usually this makes a great deal of sense as each computer is more often doing something different than any other computer connected to the Internet at any one time. But what about when 25,000 people all tune in to see the same concert or sports event? Suddenly, these people are watching exactly the same thing, but each person has to have their own individual stream. This can really slow down the Internet and diminish your experience.

Multicast takes an innovative approach to this problem. What if you created one stream, or a limited number of streams, and told every player connected to the event to read the same stream by "dipping into" it? Rather than 25,000 replications of each stream, there are only a few replications which keeps congestion low and your connection more reliable and responsive.

Multicast is automatically selected by RealPlayer if it is available for a broadcast.

Additional Information

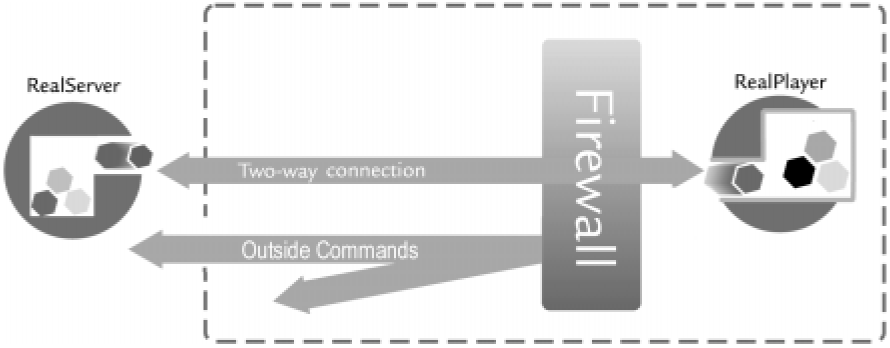
Multicast requires special hardware on the Internet that must be provided by Internet Service Providers (ISPs) and other hosting services and isn't always available. Also, Multicast may not be selected if your **Transport Preferences** are set to use only one specific transport (see Transport Preferences on page 66).

Firewalls and Security

Security is one of the biggest concerns for people connected to the Internet, whether at a business or at home. Businesses, however, have specific concerns and specific ways of dealing with those concerns. For this reason, certain precautions have become common on networks.

When you log on to a computer at work, you almost always have to type in a username and password to gain access to the full network. This is one type of security. Password security is mainly designed to control access to your machine at its present location. *Firewalls*, a different type of security, are designed to protect your computer and network from outside influences.

Figure 6-9: Firewall



Firewalls are a complex topic, but there is one specific aspect that is of interest to RealPlayer customers. In general, a firewall is a gatekeeper on the network. All outside traffic passes through the firewall when communicating with the Internet and other off-site computers. Most of this traffic is one way: the firewall will allow you to send commands out, such as requesting a Web page, but it won't allow two way communication that would give outside machines and programs control over internal machines and programs. Damage to the network and theft of information is curtailed by using this method.

However, RealPlayer requires a two-way connection to function properly. Without two-way communication you cannot fast-forward through a clip or let a RealServer know that your system can accept a higher bandwidth stream. RealPlayer can work through a firewall without exposing your network to any security risk, but you may need to change your **Proxy Preferences...** (see Proxy Preferences on page 69).

Proxy services, in concert with your Transport Preferences, allow RealPlayer to contact the Internet through a two-way connection while still providing protection to the rest of the system. Proxy services act as an intermediary, making requests on behalf of people on the inside of the firewall. For example, a web proxy knows how to get requests on behalf of a web browser. Many proxy servers/firewalls require that every program running inside a firewall have special configuration options. RealPlayer can work using proxy, but it must be configured to do so (see Proxy Preferences on page 69).

Additional Information

For a much more in-depth discussion about firewalls or to solve problems when trying to use RealPlayer from work, see our web page:
http://service.real.com/firewall/

Content: How to find it

With over eight million servers, each hosting thousands of web pages, the Internet is overwhelmingly huge. Finding content can seem daunting, but RealPlayer has several built-in features to help you, as well as access to web pages that do some of the sifting for you.

Channels and Live Stations

Your best tools are part of RealPlayer: **Radio Tuner** and **My Channels** and **Real.com Take 5**. Each have a number of pre-loaded links to clips, both live and recorded. Additionally, the Radio Tuner has the ability to search for Live Stations by content type (see Radio Tuner on page 22).

Media Access bar

Radio Tuner connects you to the latest up-to-date list of radio and TV content on the Web.

Search offers you the opportunity to type in words or phrases of interest and looks for streaming media related to those subjects (see Figure 3-10: Real.com Media bar on page 15). **Search** will return a *results page* similar to those you see when you search for Web pages. You must be connected to the Internet for **Search** to function.

Guide will take you directly to the *Real.com Guide*, the most comprehensive media hub for free Internet audio and video software and programming. Find programming featured on the Internet and play it with one click. Choose from a comprehensive listing of downloadable music, personal calendars and access to premiere programming. More than 2,500 radio and television stations, 8,000 Web sites and 500 daily live events are offered.

Web Sites Sub-Menu

The **Web Sites sub-menu** (under **Favorites**) on your RealPlayer offers everything from music to entertainment to sports to news. And because the menu talks to RealNetworks, we make sure that the connections are always the right ones. If the site moves, we keep track of it so you don't have to do it yourself.

Of special interest should be your Real.com Guide selections. The Real.com Guide is a collection of RealNetworks web pages with links to programming all over the Internet; covering live and pre-recorded streaming media of specific topics.