COMREX

IP Audio Coding

With Introduction to BRIC Technology

Introduction

By Tom Hartnett—Comrex Tech Director

In 1992, when ISDN was just becoming available in the US, Comrex published a Switched 56/ISDN Primer which became, we were told, a very valuable resource to the radio engineer struggling to understand these new concepts. As POTS codecs and GSM codecs became viable tools, Comrex published similar primers. Now, with the gradual sunsetting of ISDN availability (and the migration of phone networks to IP based services), it not only makes sense for us to introduce a product based on Internet audio transfer, but again to publish all the relevant concepts for the uninitiated.

The ACCESS product is the result of years of our research into the state of IP networks and audio coding algorithms. This has all been in the quest to do what we do best, which is to leverage existing, available services to the benefit of our core customers—radio remote broadcasters.

ISDN is not a long term solution.
The telephone network is changing.
Transition to IP is inevitable.
Resistance is futile.

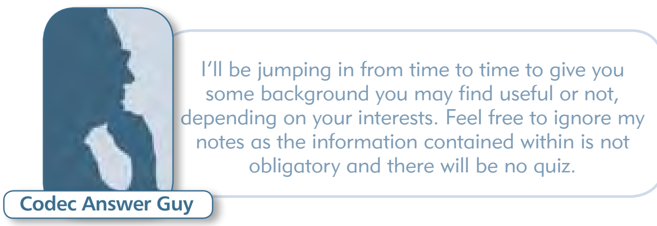
The heart of this product is called BRIC (Broadcast Reliable Internet Codec). While others have introduced hardware coined "IP Codecs," this is the first product introduced that dares to use the word *Internet* with a capital I. Given the challenges the public Internet presents, it's no small boast to say that this product will perform over the majority of available connections.

BRIC represents a change that is both desirable and inevitable for remotes. This change is inevitable because, as available connections move from old fashioned circuit switched to newer packet switched style, technology like ISDN and POTS codecs will begin to work less and less often. The desirability of BRIC stems from the new wireless networks that will make remote broadcasting more mobile, simpler and less expensive. BRIC technology has been engineered not only to be robust enough for the Internet, but usable in really challenging Internet environments like 802.11x Wi-Fi, Wi-Max, 3G cellular and satellite based Internet connections.

Remote broadcasters have been wishing for a system like this for a long time. As a former broadcaster turned designer, it is my hope that this kind of enabling technology will tickle the imagination of the user, making possible more creative and entertaining programming to be broadcast from more diverse and interesting locations. Please let us know about your unique ideas and adventures by dropping us a note at techies@comrex.com.

About IP Audio Coding

First, let's talk generally about sending audio over IP networks, like the Internet. Later we'll describe the BRIC concept and what makes it very different from IP codecs in general.



Circuit Switched Data Networks

Users of previous generation ISDN and POTS codecs are probably familiar with these basic concepts. As shown in Figure 1, these systems are based around a *circuit switched data* network (CSD).

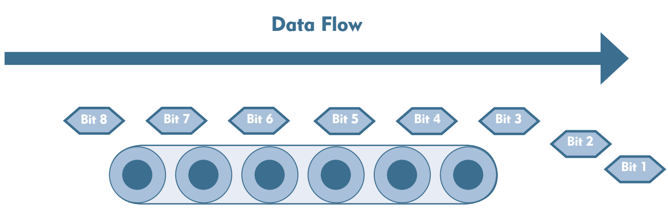


Figure 1- CSD Network

CSD networks can be treated like conveyor belts in that data is applied to them at a constant rate and defined order. While a connection is maintained between the two points, data will continually and reliably fall off the far end of the conveyor belt at precisely the same rate and in the same order. The speed of the conveyor isn't fast enough to deliver raw, uncompressed digital audio, however. So a compression encoder and decoder are applied at their respective ends of the link, achieving a data rate that matches the conveyor belt perfectly. This is shown in Figure 2.

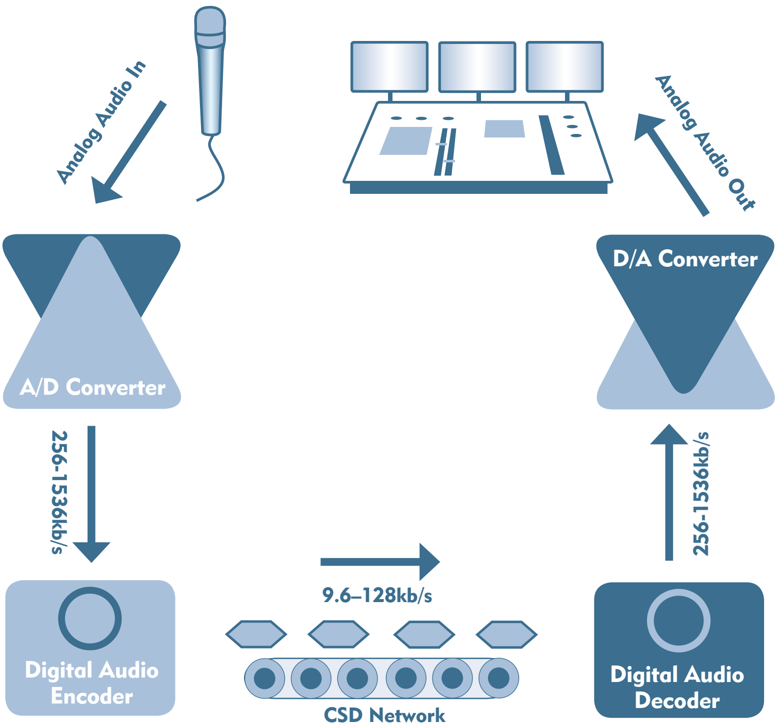


Figure 2- Compressed Audio Over a CSD Network

For clarity, only one direction of data flow is shown. Since most CSD networks are full-duplex, this process is generally repeated in each direction.

One final point to make about these codecs is that the encoders and decoders typically don't process digital audio in a bit-by-bit (or digital word-by-word) fashion. Most encoders take a discrete "window" of input audio before beginning processing. This snapshot-in-time of the audio is known as a *frame* and can range from 1/50th second to several hundred milliseconds, depending on the codec algorithm. So while the network "conveyor belt" is handling a steady stream of data, the encoder is buffering and outputting a full frame at a time, and the decoder is buffering and inputting a full frame at a time. This accounts for much of the audio coding delay experienced by these systems.

IP Networks for Audio

IP networks (like the Internet) fall under a category known as *packet-switched* data networks. As shown in Figure 3, rather than a single connection between two points, IP networks rely on multiple "relay stations" between their points, known as *routers*. Data is bundled together in *packets* and sent into the network with a source and destination address attached as an *IP header*. This header acts much like the address on a mail envelope, being read by the various routers in the network and relaying the packet along one of many possible routings. Each packet may receive a completely different routing through the network. No actual connection between the endpoints is created, but the constant flow of packets between two points creates a *virtual connection* or *stream*.

Codec Answer Guy

In reality, each of our packets requires three separate headers (IP, UDP and RTP) containing a multitude of information, some useful and some not. But the pesky Internet standards require that routers see it all and will complain loudly at their absence, dropping your valuable packets into the infamous and evil "bit-bucket."

The source and destination addresses in an IP header are binary numbers 32 bits long, and each device connected directly to the Internet must have a unique one.

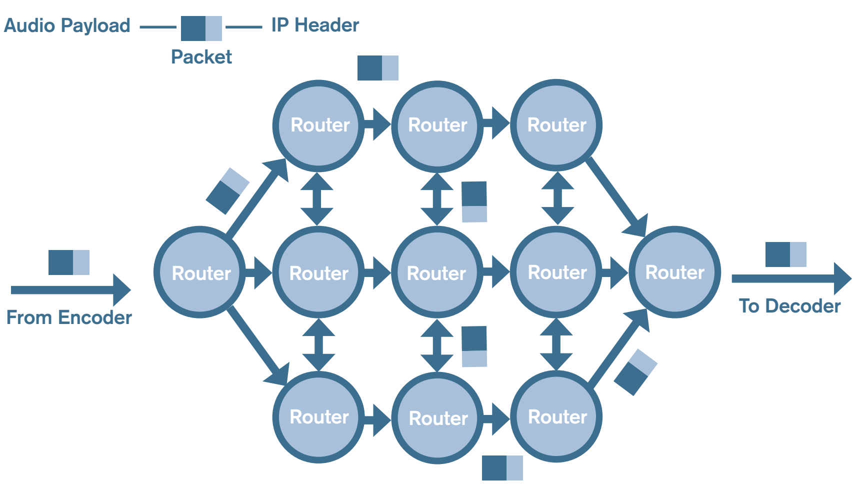


Figure 3- IP Network

To express IP addresses, we usually segment these 32 bits into four 8 bit numbers and express each of these in decimal (from 0-255) with dots between them. E.g. the IP address of the Comrex ACCESS "test line" is 70.22.155.133.

Codec Answer Guy

That's 01000110000101101001101110000101 in binary…see why we abbreviate it? By the way, we've completely skipped the fact that the inter-connections to and between these routers may be physically different; some may be wired or wireless, and most are packet-based protocols riding on top of circuit-switched physical links like T1 phone lines. In fact, from here on we blissfully ignore all this and treat the network as a cloud, implying a dream-like state of ignorance.

IP packets may be virtually any length, but as mentioned earlier our encoders typically deliver fixed length frames of audio data. In order to keep to the minimum possible delay, it makes the most sense to include only one audio frame per packet. This way the encoder delivers an audio frame to the network the instant it is ready. If the audio frames are very small (such as in an encoder with a very short frame and very high compression) the IP header can actually rival or exceed the size of the audio frame. In these circumstances there may be an advantage to clustering several frames into a packet, since it will reduce the overall bandwidth requirement of the network (but lengthen the audio delay slightly).

Referring back to CSD, we noted that most CSD networks provide a return channel automatically, resulting in full-duplex communication. Since no channel is actually created in an IP connection, it follows that no reverse channel exists either. IP codecs will typically try to simulate a full duplex channel by creating a virtual stream in the opposite direction, but this is completely discretionary and may be disabled without causing any performance issues.

Bandwidth in IP

On CSD networks, the speed of the "conveyor belt" dictates the exact digital bitrate that is available between the two points. On modem connections (like in POTS codecs) this is usually between 24-56kb/s, while ISDN offers speeds of 128kb/s. IP networks aren't specified in terms of data rate but rather a *maximum* rate, which is often an order of magnitude higher than CSD networks — usually in the Mb/s range. Given this high capacity, why bother compressing your audio data at all?

Well, for IP networks that you control, the answer is simply "don't compress." But we're primarily focusing here on networks out of the control of the user. *Managed* networks like LANs and some types of WANs allow you to prioritize and control *Quality of Service* (QoS) between different types of traffic. But the public Internet has no QoS available, and all traffic is usually treated equally. So you pretty much have to send your packets into it and cross your fingers and hope for the best.

Figure 4 shows a typical Internet connection. User A and User B are trying to connect to each other, and each has a fairly fat pipe to the Internet. But each of these pipes opens to a LAN environment, where it is shared between several users on each premises. And each of these pipes terminates at an Internet Service Provider, which proceeds to lump each of their fat pipes together through a sharing router to a single Internet link. Finally, once delivered to the Internet, each connection must share the routers and the inter-router links with a multitude of other users, resulting in contention and overall Internet congestion. The end result is that the actual throughput between A and B is undefined, changing constantly based on the usage patterns of other connections.

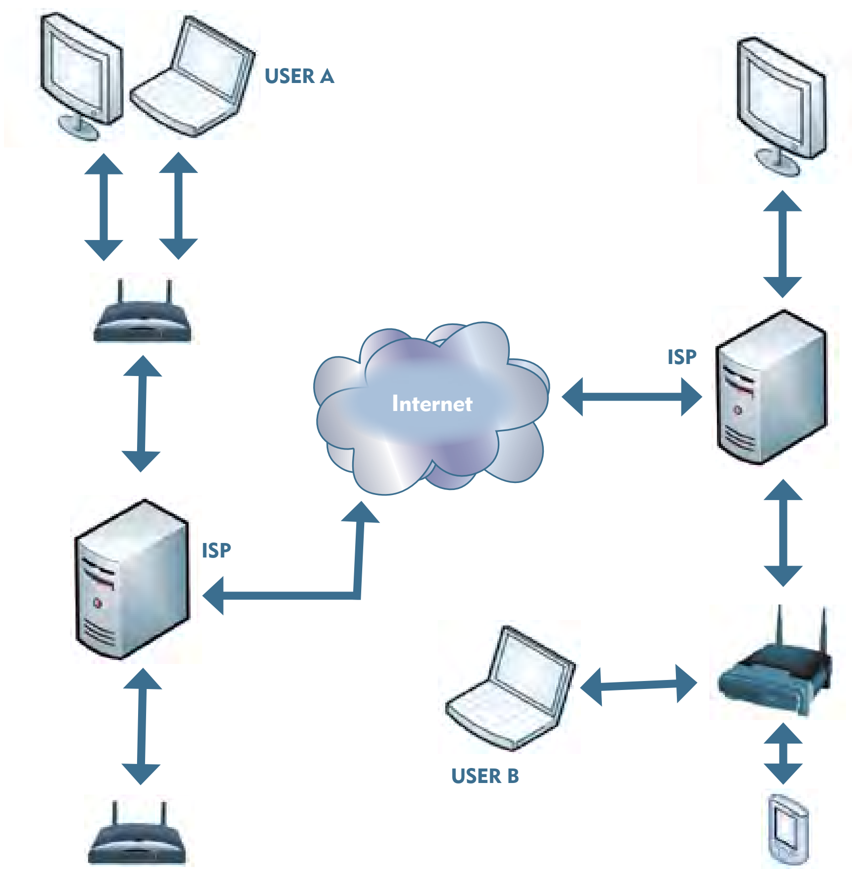


Figure 4- Typical Internet Sharing

The simplest and most powerful way to assure that a congested network still performs well with IP audio is to demand as little as possible from the network. This is why we need to compress. While you may achieve multi-megabit connections for a short time, your audio transfer is only as good as the worst connection experienced. You'll read more later about how the aggressive compression modes offered by BRIC can result in highly stable connections.

There are other ways to convince an IP network to deliver packets quickly and reliably. One is to send your packets more than once. If your Internet pipe is indeed fat, you can enhance the reliability of it by introducing Forward Error Correction (FEC). In its simplest form, each packet sent contains a copy of one or more previous packets, so that any packets that are lost or delayed may be reproduced. This is shown in Figure 5. More advanced FEC provides for sending special parity packets that allow lost information to be recovered through special decoding algorithms. In all its forms, however, FEC requires higher digital bandwidth than the raw data and may not be feasible on all networks.

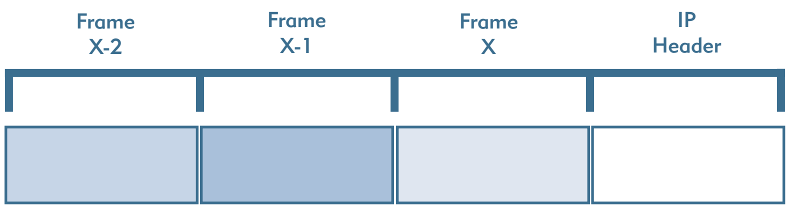


Figure 5- FEC by Resending Packets

In reality, it's actually pretty rare for a packet to be completely lost on the Internet, and it is much more likely that some packets will be delayed in arrival. A static network delay is fairly easy to deal with, since all packets will arrive in relatively good order—all delayed by the same amount. But this is not usually the case with the Internet. With constant variations in traffic load and routing, it's much more likely that some packets will arrive late, while others will arrive early. Or possibly, packets will arrive in large bursts with long lags of nothing in between.

In an Internet audio stream, the difference between the earliest arriving packets and the latest arriving packets is known as *jitter*. There's only one way to fix jitter, and that involves building a buffer of incoming packets at the decoder to smooth out the incoming jitter. This, of course, means adding a delay to the link, which for our purposes is a very bad thing.

A jitter buffer is shown in Figure 6. The bursty, out-of order packets enter from the left, and play out smoothly to the right. The longer the buffer is, the more likely the decoder is to receive an uninterrupted data stream.

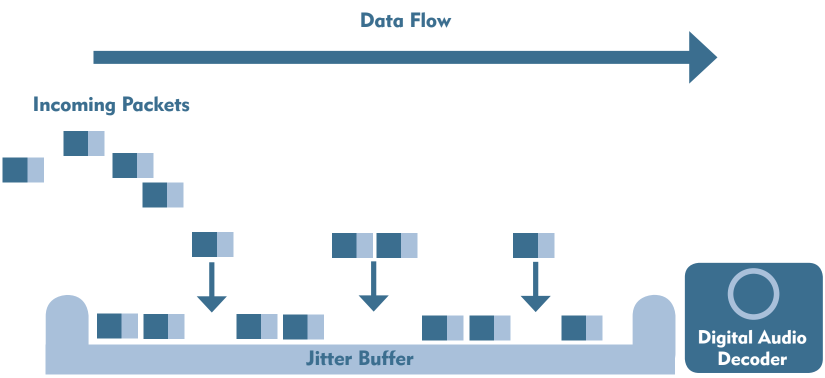
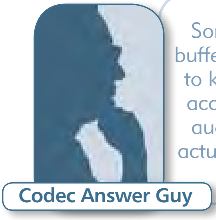


Figure 6- Decoder Jitter Buffer



Some of the "magic" behind BRIC is in how this buffer is managed. It's vital to keep the buffer small to keep delay at a minimum, but large enough to account for large changes in jitter. Because BRIC audio coding algorithms are so resilient, we can actually survive a fair amount of packet loss before having to increase the jitter buffer.

In summary, the basics of an IP audio system are shown in Figure 7 and consist of

1) An analog/digital converter followed by a compression encoder

2) An IP packetizer and transmitter

3) An IP packet receiver and decoder jitter buffer

4) A compression decoder and digital/analog converter

These pieces build an IP audio codec that can perform well on networks that are tightly controlled, such as managed LANs or WANs. But often they perform poorly in the "wild" of the public Internet. Now that we've covered the basic aspects of all IP audio devices, let's take a look inside the BRIC concept to see what is different and special about it.

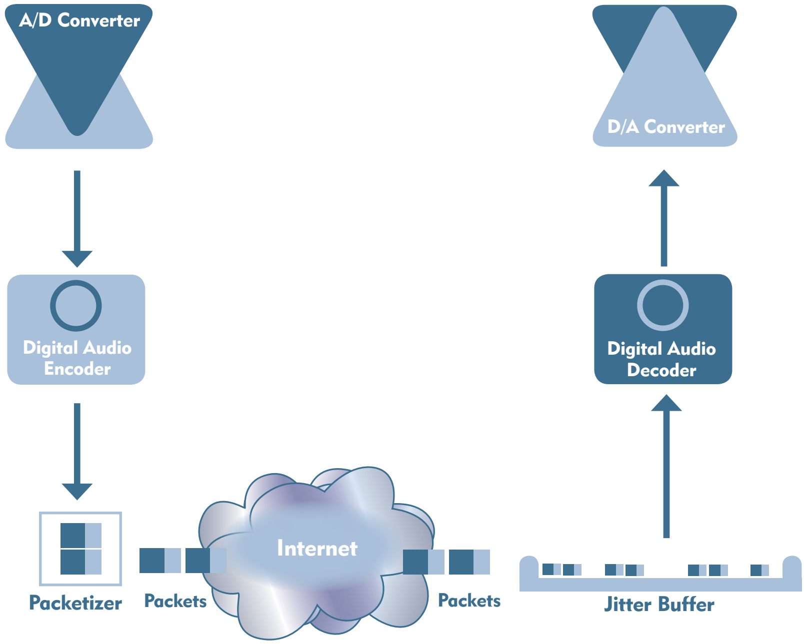


Figure 7- IP Codec Building Blocks

BRIC

The BRIC (Broadcast Reliable Internet Codec) actually goes beyond IP codecs in five ways:

1) BRIC utilizes powerful algorithms which can achieve remarkable compression, delivering high quality audio in a trickle of digital bandwidth.
2) BRIC coding algorithms have very low delay, allowing full-duplex communications with ease.
3) BRIC algorithms are resistant to packet loss; very effective error concealment is used to hide lost information.
4) BRIC uses intelligent jitter buffer management protocols, coupling with the packet loss resilience to keep delay to an absolute minimum.
5) The BRIC Traversal Server (TS) allows location and connection to other users on the Internet, even those not usually visible from the public Internet.

BRIC Algorithms

BRIC offers three default coding modes:

BRIC-ULB— (Ultra Low Bitrate) This mode is designed for speech transfer only but has a remarkable compression ratio of around 25:1 allowing it to deliver 7KHz audio at an astonishingly low bitrate (around 14 kb/s). The voice quality of BRIC-ULB is comparable to the old codec standard G.722 (7KHz), but uses less than 1/4th the amount of data.

What gives BRIC-ULB its power is the fact that it's a variable rate codec. The BRIC-ULB encoder actually varies the size of the packets it delivers to the Internet based on two factors as shown in Figure 8: First, it bases its overall output frame size on the complexity of the input audio during that frame. Periods of silence get tiny packets, and complex input gets full frame packets. Secondly, the encoder accepts network congestion information from the decoder and uses this to base decisions on packet size within the confines of the audio complexity factor.

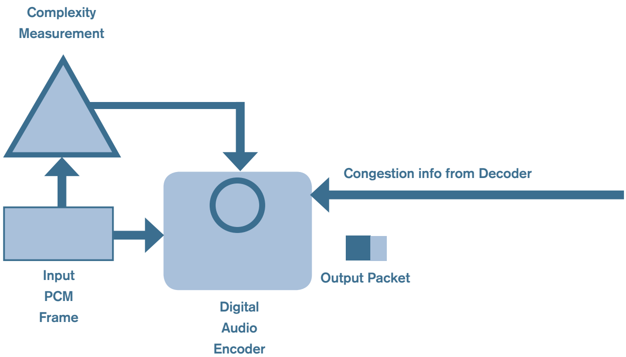


Figure 8- BRIC-ULB Variable Bit-Rate Decisions

Like all audio codecs that have this level of effectiveness, BRIC-ULB processes audio in frames. But these frames are kept extremely short, and an entire encode/decode cycle is achieved in less than 1/10th second. This means that you can carry on a full-duplex conversation, using BRIC-ULB, with ease. Even on high throughput networks, BRIC-ULB provides enough bandwidth conservation to allow use of FEC to increase reliability and reduce network delay on the link. BRIC-ULB may also be used as the reverse channel for BRIC-HQ2 to reduce round-trip delay.

BRIC-HQ1 (High Quality 1)— This mode keeps delay low (around 1/10 second) but allows for full fidelity (15KHz) audio transmission. It delivers music or voice audio equally well and runs at a data rate of 28kb/s. This mode also allows "dual mono" transmission, so that two independent audio signals (using twice the network bandwidth) can be sent to the same location. This algorithm may be used to connect to earlier Comrex POTS codecs (such as Matrix, BlueBox or Vector) on a standard phone line.

BRIC-HQ2 (High Quality 2)— This mode minimizes artifacts and encodes speech and music equally well, and provides a 12 or 15KHz fidelity signal over 24kb/s of network bandwidth. BRIC-HQ2 also allows for stereo operation at the lowest data rate of 24-30kb/s, making stereo over a single modem connection possible. BRIC-HQ2 has moderate delay of about 1/3 second, which is in the range of many ISDN codecs.

BRIC-HQ1 and **BRIC-HQ2**— Operate at data rates typical of modem connect speeds and less than half the rate of a single ISDN B channel. They keep their digital bandwidth low enough that the data still fits comfortably into the upload channel of most 3G cellular systems and can be used on Wi-Fi and other challenging networks easily.

Codec Answer Guy

Comrex ACCESS codecs provide two additional coding algorithms that, while requiring higher data rates, deliver near-transparent audio for critical listening applications. HE-AAC (comparable to AAC+) is useful on links where delay is not a concern. AAC-LD is available for delay-critical applications that demand pristine audio—as long as reasonable network bandwidth is assured. Golden ears rejoice!

Error Concealment/Jitter management

One of the most challenging tasks of an IP codec is determining how much to extend the jitter buffer. This choice will have a dramatic impact on overall system delay and stability. It's vitally important to find the "sweet spot" where a perfect balance between packet loss and delay exists.

BRIC algorithms are much more resilient to the effects of packet loss than most other algorithms. BRIC has built-in concealment algorithms, inducing noise or packet repeats as needed to keep these losses unnoticed. Since more packet loss can be tolerated, the jitter buffer management algorithm can be fine-tuned to minimize delay.

When jitter buffer correction is required, the buffer management routine kicks in to make small, barely noticeable changes to the decoder playout rate to allow for the new buffer conditions. If a dramatic change in incoming data rate occurs, the jitter buffer is extended very quickly, then very gradually retuned to a lower value based on recurring error-free reception.

Codec Answer Guy

Comrex ACCESS codecs give an easy-to-follow visual representation of the jitter buffer, as well a historic plot of packet loss. These can be used to override the default settings to manually make buffer length, packet size, and FEC adjustments. Absolute paradise for those compulsive knob twiddlers!

BRIC Traversal Server (TS)

The last piece of BRIC (no pun intended) is something that is not located at either end of the link, but exists on the public Internet and is maintained by Comrex. BRIC TS acts as a directory to all BRIC-based codecs that are attached to the network. Its use is optional, but if enabled it offers the following services:

1) Creates a "Buddy List" of other interested users and allows quick and easy connections

2) Provides NAT Traversal

Buddy Lists

Once you have established a BRIC TS account and logged on to the server using a standard web browser connected to the Internet, as in figure 9, you can easily subscribe to BRIC TS user lists, and allow others on that list to see your status and location. This information is reflected in the user interface on each codec, as shown in Figure 10. To connect to a buddy, simply point and click.

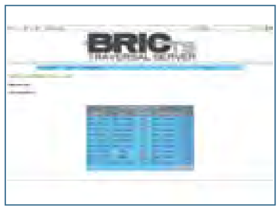


Figure 9- Traversal Server

Figure 10- BRIC-TS enabled ACCESS Portable

NAT Traversal

First let's talk a little about Network Address Translation. Although we mentioned earlier that every device connected directly to the Internet needs a unique IP address, it isn't practical to assign every computer its own address. (After all, there are only around four billion addresses available!) So NAT provides a way to share a single address between multiple users. NAT is built into virtually every router available and provides for a rudimentary firewall, since unsolicited incoming traffic can't get through. A typical NAT arrangement is shown in Figure 11.

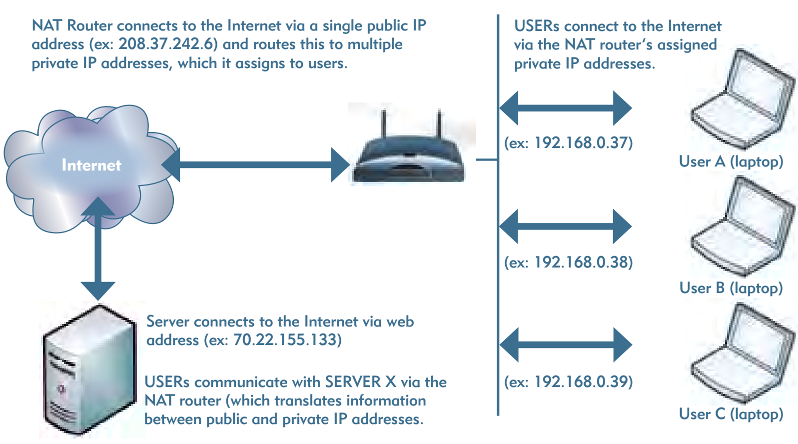


Figure 11- Typical NAT Scenario

In the scenario shown in Figure 11, user A is assigned a local IP address by the NAT router, but the router has its own address which it uses to connect to the Internet at large. When user A sends a message to server X out on the Internet, the router translates the "source IP address" and sends it on. Since it is aware of the fact that an outgoing message was sent, it is prepared to forward the response from server X back to user A. This is known as creating a NAT tunnel, or path, that will allow free flow for a limited time. But if anyone else tries to address user A, data is blocked because the NAT router has not opened any tunnels for them to utilize. NATs can even be tiered, with one NAT device translating for a second NAT device, and so on.

Most home Internet users connect through NATs. Also virtually all Wi-Fi and many other wireless connections are NAT oriented. But NATs have one drawback; if your codec is on a network that is protected from unsolicited outside messages, how can anyone make a connection to you?

The answer is they can't (well, not without help anyway). But if only one of the two BRIC codecs in the link is behind a NAT, connections are easy as long as they are initiated from the NAT codec, as shown in Figure 12. The router will accept the outgoing request from its codec, and open a tunnel in the other direction. So by putting your studio codec on a publicly visible IP address, you can be assured of making incoming connections from the vast majority of codecs out there without any help from BRIC TS.

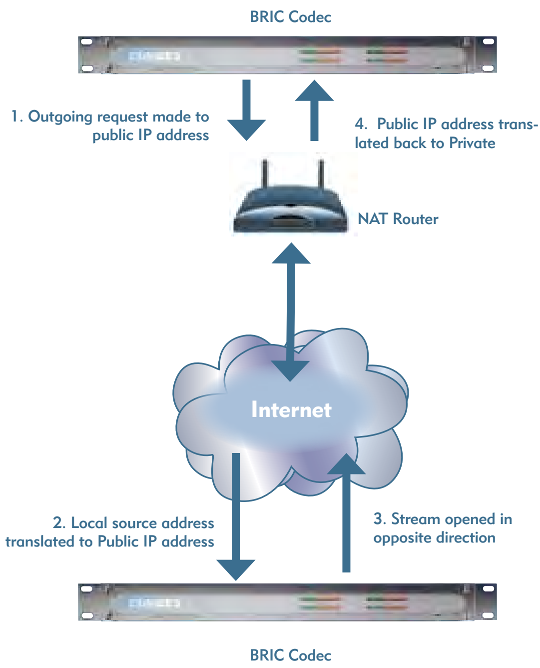


Figure 12- Connection from Behind a NAT

Sometimes NAT connections can't be avoided however, and that's where BRIC TS helps. Figure 13 shows two NAT users who wish to connect to each other. In this scenario, a BRIC TS subscription has been enabled on each unit. The BRIC TS server has kept a constant tunnel open through each NAT and has a record of where each unit is located. When user A wants to connect with user B, he simply inquires about user B's location on BRIC TS. BRIC TS can then inform user B (through the tunnel it has maintained) that user A would like to connect. Users A and B can now send outgoing messages to each other, opening a tunnel between them through their NAT routers.

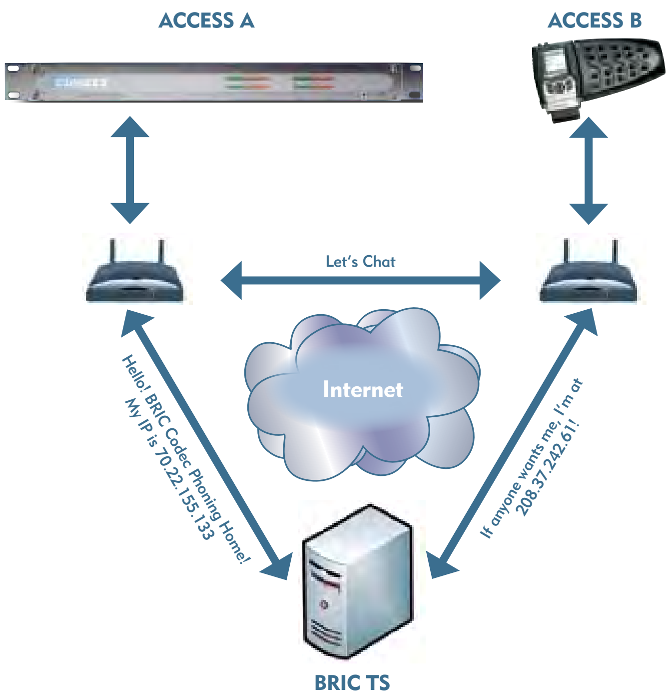


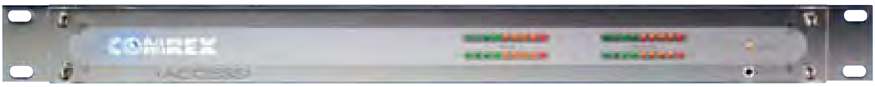
Figure 13- Using BRIC TS to Communicate from Behind NATs.

Use of BRIC TS requires a license unlock for each BRIC-enabled device that you want to access via the service. Once the license has been activated, you'll be given a user name and password for the BRIC TS server. Please contact Comrex for more details on how to activate BRIC TS for your BRIC-enabled devices.

Codec Answer Guy

Comrex ACCESS Codecs

The first embodiment of BRIC is the Comrex ACCESS codec. Like most broadcast codecs, it is available in rack-mount and portable versions that carry different options.



Comrex ACCESS Rack

ACCESS Rack is designed as the "set and forget" end of the IP link. It has no hardware user controls on the front panel. Connection and management are done over a LAN using the intuitive web page served from the unit. This way, ACCESS may be controlled from any computer using a normal web browser.

Some features of ACCESS Rack are:

- Uses BRIC technology to deliver broadcast audio over the public Internet
- Input and Output level meters on front panel
- Audio level meters also available via web page interface
- Connections via Ethernet and Modem jacks
- Stereo analog in/out on balanced XLRs
- AES3 Digital audio connections
- Connections for serial ancillary data and four contact closures
- Connections for keyboard and monitor to allow initial configuration
- AAC, HE-AAC (comparable to AAC+), HE-AAC v.2, AAC-ELD and AAC-LD modes available as options
- Backward compatibility to Comrex POTS codecs

The ACCESS Rack User Interface

ACCESS

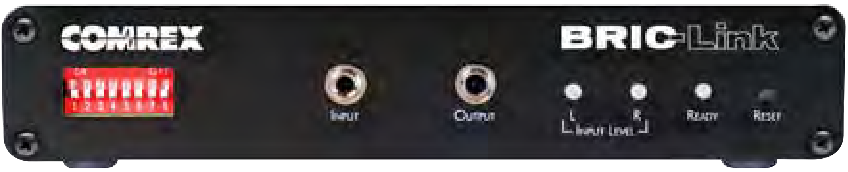
ACCESS Portable

ACCESS Portable was designed as a remote broadcasters dream. About the size of a camcorder, the hand-held unit may be powered for seven hours by its rechargeable battery. For broadcasts that require larger mixing and headphone capabilities, the hand-held unit may be "docked" into the stereo mixer/headphone management section to provide a full featured, table-top, 6-channel mixer/codec.

Some features of the ACCESS Portable are:

- Uses BRIC technology to deliver broadcast audio over the public Internet
- User interface via integrated LCD/touchscreen display menus
- Web browser included for connection to Wi-Fi access points that require log-in
- Built-in Ethernet port
- Built-in rechargeable battery supplies 7 hours of talk-time
- Connection to a range of other networks via integral Cardbus slot including
    - Modem Connections
    - 3G Cellular Links
    - Wi-Fi cards
    - USB port for certain 3G modems or USB keyboard
- Ethernet port acts as Internet sharing device, allowing use of laptops on circuits utilizing Cardbus cards.
- AAC, HE-AAC (comparable to AAC+), HE-AAC v.2, AAC-ELD and AAC-LD modes available as options.
- Backward compatibility to Comrex POTS codecs

BRIC-Link



- Designed for use on dedicated data circuits such as T1/E1, WAN, LAN, ISM Band IP Radios and satellite data channels
- Fully bidirectional mono and/or stereo transmission
- BRIC Technology for ultimate stability and minimal delay
- Performs well on dedicated public Internet connections using AAC compression modes
- Includes FLAC lossless compression, AAC, HE-AAC and linear modes
- Features Streaming Server Mode for use with third-party Media Players such as VLC and WinAmp using HE-AAC
- Fast, easy setup via networked Windows PC application
- Simple web browser control interface

Summary

While we've shown you the building blocks of an IP codec, it takes more than that to provide a system that will transform unreliable Internet connections into reliable broadcast connections. BRIC enables this by delivering powerful and robust audio coding, intelligent connection management and ease-of-use. BRIC TS goes a step further in making the entire system plug-and-play by offering point and click connectivity and Traversal functions that allow connections to users in closed LAN environments.

Comrex ACCESS codecs leverage the power of BRIC and provide broadcast-friendly features like balanced I/O, AES3, battery power and mixing functions into rackmount and portable road-ready packages.

We hope you've found this primer useful, and we'll have lots more to offer on our website. Some topics that will be covered there include:

- 3G cellular data capabilities and worldwide availability
- Experiences using public Wi-Fi for remotes
- Product updates and new features
- BGAN and VSAT satellite applications

Please check it out all the latest at www.comrex.com.

Put Comrex On The Line.

Toll Free in USA: 800-237-1776 • www.comrex.com
e-mail: info@comrex.com
19 Pine Road, Devens, MA 01434 USA
Tel: +1-978-784-1776 • Fax: +1-978-784-1717

COMREX