Redefining Broadcast Transmission Systems: A Software-Based Approach

By Ian Fletcher and John Wadle

Ian Fletcher

John Wadle

With the continued advances of information technology into content management and production workf ows, broadcast facilities are becoming largely information technology (IT)-centric operations, leveraging proven enterprise-class server and network technologies. A major exception to this trend, however, is the broadcast transmission process (master control and playout), which remains heavily dependent on specialized broadcast hardware. In this process, transmission of a moderately complex television channel can involve a dozen or more specialized hardware components linked by an automation system. In many cases, the costs associated with this traditional approach to broadcast transmission can be prohibitive for internet protocol television (IPTV) and other new operators intending to set up and manage their own playout facilities. Furthermore, to compete head-on with mainstream broadcast channels, new service providers need to match the quality standard and aesthetics that an increasingly discerning audience has come to expect. The ability to move the transmission process to an IT platform opens the door to an entirely new business model for broadcasters. With a fully-featured broadcast transmission chain provided by a software solution operating on industry-standard IT servers rather than specialized broadcast hardware, it is now possible to build and operate a broadcast transmission center as an extension of an IT-centric enterprise.

Broadcast Engineering Challenges

The inroads already made by information technology (IT) into broadcast operations mean that most modern broadcast facilities incorporate extensive internet protocol (IP) networks, and many also include large digital storage resources supporting file-based production workflows. However, as described above, this transformation to IT has stalled at transmission suite where legacy workflows and the need to accommodate traditional methods of content acquisition and distribution rely on costly, specialized broadcast hardware.

In broadcasting, the advance of new technologies and the demand for new services have been—and continue to be—relentless. In this context, broadcasters look for applications of technology that allow them to build increasingly advanced production and transmission facilities with higher degrees of automation and more sophisticated workflows.

Moreover, the rapidly evolving market for multimode distribution demands more channels across a wider range of delivery platforms at a lower cost per channel.

Until now, the broadcasters' response to these demands has been to seek increasingly complex products from manufacturers of broadcast equipment and automation systems. Using a mix of equipment and software from multiple vendors, a "best-of-breed" solution can be assembled by each individual broadcaster for each requirement. These complex systems attempt to provide a fully integrated solution from a number of discrete parts. However, with this approach, lowering the cost per channel remains an elusive objective.

It's time for a change. Looking at a typical, automated broadcast transmission chain will help in understanding the details of this change (**Fig. 1**).

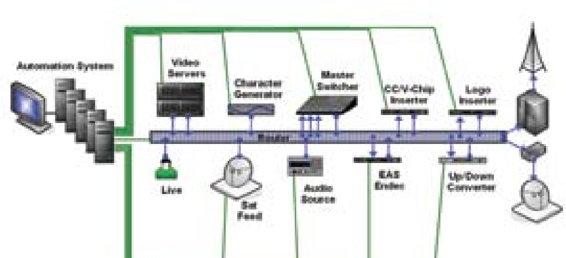

Figure 1. Typical broadcast transmission chain with automation.

Each device in the transmission chain must be separately connected and controlled by the automation system using the unique protocol of each device. Sometimes this control uses a network connection (i.e., Ethernet), but in many cases devices are controlled via serial communication (e.g., RS-422). During playout, a complex interstitial sequence managed by automation can require actions by six or more broadcast devices controlled by one or more automation computers. For a single program transition, dozens of independent messages must flow among these computers and the broadcast devices, with all message transmission and resultant processing occurring within a single video frame.

In addition to this level of complexity and rigorous timing constraint, there is the further challenge of keeping such a sophisticated system reliable. For example, updating a manufacturer's firmware for a single device can cause problems for the automation system and affect the entire chain. This is a complex and expensive system to operate and maintain, because of the quantity and diversity of equipment involved.

With this model, the costs of owning and operating a fully featured transmission chain for each channel can be prohibitive, particularly for new players in the market. It also becomes expensive and operationally complex to launch new services, or to add graphics and branding to existing services—both requiring the introduction of additional hardware components.

Facing this cost, some new service providers resort to "cuts only" linear playout using only a router and video server. The resultant channels have no transitions effects between segments, and no provider branding, and are unable to overlay inapplicable logos embedded in the recorded video.

This approach may have been acceptable for new distribution mode services in the past, but with today's fierce competition for viewers, the creative elements of a broadcast channel have become increasingly important. Sophisticated program presentation, with aesthetically pleasing transitions, animated branding graphics, and creative promos between programs, helps to differentiate one channel from another, builds brand identity, and creates viewer loyalty.

For newer services, such as IPTV and mobile delivery in particular, it is not sufficient simply to stream clips to home computers, set-top boxes or handheld devices. Despite the widespread use of home computers and handheld devices, consumers view any content delivery going on behind the glass as "television" and they judge the quality of delivery along with "normal," over-the-air or cable television channels. Consciously or not, a viewer will tend to switch away from any channel that does not have the quality they have come to expect.

The Technical Challenges

The challenges involved in delivering a broadcast channel via a software-based transmission system, while providing the same level of quality and reliability provided by specialized broadcast hardware and traditional automation systems, are substantial.

A few years ago, Omnibus Systems began a research and development project (Project 9) to create a software-based transmission system operating on standard IT hardware. The goal of this project was to provide a viable alternative to the traditional automated broadcast transmission chain. It is well known that computers can play recorded video and audio, and with the addition of specialized video output cards, high-spec IT servers are quite capable of basic playout in broadcast standard signal formats such as SDI/601 or HD-SDI.

The objectives of the Project 9 software included:

- Support for multiple video compression formats, including MPEG-2, MPEG-4/H.264, Windows Media/AV1, DV25/50, and MXF, as well as realtime decoding of these formats.
- Support for the full range of video resolutions and aspect ratios required for SD and HD television, IPTV, and mobile device streaming services.
- Realtime up, down, and aspect ratio conversion during transmission, based on the requirements of the channel.
- Realtime video transition effects with quality equivalent to broadcast hardware.
- Two-dimensional (2D) DVE moves for picture-in-picture (squeezebacks) with still frame backgrounds.
- Support for multiple audio tracks and 5.1 surround sound, including AC3/Dolby-D and Dolby-E.
- Sophisticated audio transitions including voice-overs and audio lead-lag.
- Multiple, simultaneous linear keys with in-out transition effects.
- Animated graphics for overlays, using .GIF or Flash.
- Character generator tool for creation of multipart graphics with text inserts.
- Preservation of ancillary (VBI/VANC) data throughout the video process.
- Realtime insertion of VBI/VANC data for captions, subtitles, V-Chip, etc.

Microsoft Windows XP Professional and Windows 2003 server were chosen as the client and server operating systems for the project.

It was decided that AMD processors and their associated chipset architecture were particularly wellsuited to the graphics-intensive operations that were required for mixing high-definition video.

The A/V processing software for the project was designed using Microsoft .NET and the DirectShow Framework. This enabled maximum advantage of using available technology such as video encoding and decoding software. The .NET and DirectShow architecture also provided a "plug-in" model to accommodate customization to the specific needs of each broadcaster.

To generate a broadcast-standard output signal, the servers were equipped with video cards to produce a standard serial digital video output signal. These cards do no video processing, but simply convert the YUV colorspace bitmap produced by the software to a 601/SDI or HD-SDI signal, as required.

The first development challenge was to achieve reliable, full-resolution video playback. This required deterministic throughput for every video frame from content storage, through the server, and out the video card, without delay or drop in frames due to processor usage, disk storage bandwidth, or network activity. To achieve this, a unique buffering methodology was developed—upstream of the software modules for decoding, mixing, and overlaying the source content—coupled with a frame-accurate, time source locked reference acting as a master clock for the entire software process.

The next and most complex phase of the project was the development of a software A/V mixer that would enable all required video and audio transition effects to be performed, as well as 2D DVE operations.

Starting with a basic video mix (cross fade), and utilizing the sheer horsepower of the AMD Opteron dual-core processors, two full HD images were blended together in less than 30 msec. This initial step was within the time window of a single video frame, but did not provide enough headroom for reliable performance and other concurrent operations. However, by making use of the Digital Signal Processing (DSP) instruction set provided by the Opteron processors, there was significant improvement in this performance and the average time for this HD mix was reduced to less than 2 msec.

As a multitasking environment, the Windows Server operating system is not designed to provide a guaranteed level of service. Therefore, to achieve the performance needed for a realtime broadcast system, a significant amount of development time was expended perfecting time-sensitive thread control within the Project 9 software. This mechanism ensures that the transitions between consecutive video clips are smooth, and that the next sequential video frames/fields are always available when needed.

A similar technique was used to provide independent control for overlays such as linear keys and voice-overs. Here, the software-based video overlay process easily surpassed the capabilities of the hardware model. For example, the resultant software-based linear keyer has no limits on the number of concurrent keys or logos that can be cut into the video at one time.

The software process for decoding, mixing, and overlaying two content sources is similar to the one shown in **Fig. 2**. This high-level view of the A/V process illustrates how the functional equivalent of a hardware-based transmission chain was constructed in software.

As shown in **Fig. 2**, video and audio effects require that at least two content streams be managed simultaneously. In the early stages of the project, all streams were sourced from compressed digital content files held in storage accessible via the local area network (LAN). These files are moved to the server in large "chunks" and buffered, before being passed to the first step of

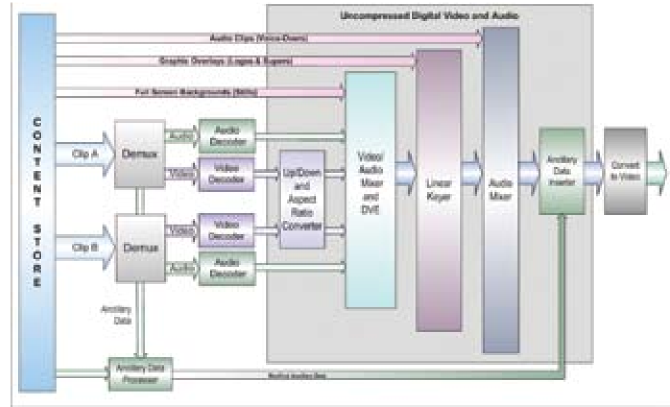

Figure 2. Software A/V process architecture.

the software A/V process (demuxing), which splits the content stream into its component parts. The resultant compressed video and audio streams are then passed to appropriate software decoders. The decoders chosen for the project use the plug-in architecture of Microsoft DirectShow, thus providing the ability to select from a wide variety of available third-party decoders for different compression formats.

After decoding, the video image is scaled/sub-sampled into the correct size and shape for the defined aspect ratio and pixel resolution of the channel. This can range from quarter source input format (QSIF) and source input format (SIF) for mobile devices right up to full HDTV at 1920 x 1080. Obviously, realtime rendering of video mixes at HD resolutions requires much more processing power than SD or mobile-quality video and demands the power of the Opteron dual-core processors.

Video Mixing

The video mixing process is the heart of the system and uses the processor's DSP instruction set to perform highly optimized, full-screen transitions, including all the standard master control effects.

The video mixer module is also responsible for doing 2-D digital video effects (DVE) moves and provides independent sizing and positioning of two channels of full-resolution video over a background graphic (i.e., picture-in-picture). One unexpected benefit found in moving to a software-based DVE is that the process does not introduce the frame delays that are common to hardware-based DVEs.

Audio Mixing

Another area in which the software-based process works well is audio mixing and effects. Instead of simply having the audio follow the video on transitions, a more sophisticated level of audio transition control that is increasingly demanded by the broadcasters can be provided. This includes independent control of timing incoming and outgoing audio levels, in other words "lead and lag" capability (**Fig. 3**).

Realtime audio level control becomes a simple task in software, and in the context of automated playout, also offers

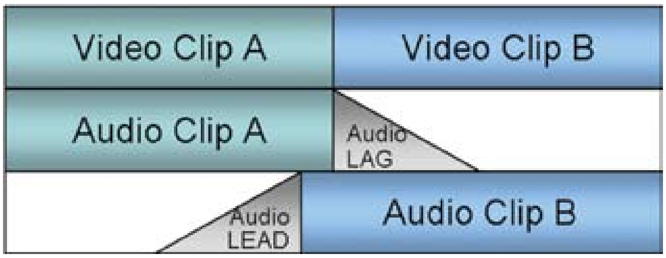

Figure 3. Audio transitions: lead and lag.

the capability of adjusting levels on individual clips in the playlist—a feature not seen before in broadcast automation.

Linear Keys

The next stage in the video process is linear keying. This software keyer module allows any standard graphics image to be resized, positioned, and used as a logo. If the image contains transparency data, these are used to blend the key over the underlying video stream.

Again, the software DSP processing works well here, so the addition of multiple key layers has little impact on the overall processor utilization of the server. Tests have shown that it is possible to cover the entire video stream with logos and still maintain smooth playback.

Because animated logos are becoming extremely common and animated .GIF files can be produced with many standard software applications. The project software was designed to handle these with automatic looping to provide continuous animation of the key.

Voice-Overs

Although the architecture of the A/V mixer software already supported basic audio mixing, the decision was made to follow the model of a broadcast transmission chain and provide an additional downstream audio mixer for the insertion of recorded voice-overs. These can be provided in any of several audio file formats including WAV, MP3, and WMA. The voice-over module provides the ability to adjust the level of background audio suppression as well as the speed of the transition ramps.

VANC Data

The final stage of the A/V process is the re-insertion of any vertical blanking interval (VBI) or vertical ancillary (VANC) data that was captured at the demux point. These include data for closed captions, V-Chip ratings, and other such functions. There is no mechanism for handling these data as part of video mixing operations; thus the VBI/VANC data that are extracted during the demux process are held aside during the mixing operations. These data are then frame-accurately re-inserted at the final output stage.

In addition, because of the plug-in design of the overall A/V processing software, a plug-in module concurrent with the A/V mixing process can perform modification or insertion of VBI/VANC data. This enables the insertion of additional VANC data—such as V-Chip—in realtime during transmission. The flexibility of this software model makes it very simple to add channel-specific VANC data without the need to add downstream hardware.

Live Sources

To accommodate the requirements of many broadcasters, the software is also needed to accommodate live sources such as news studios or network satellite feeds. To enable this, the video output card chosen for the project servers included a 601/SDI or HD-SDI input in addition to an output in the same signal formats.

With this simple addition and relatively minor enhancements to the A/V process, it was possible to handle a live source as either primary or secondary content, while retaining all the capabilities already in place for video mixing, audio mixing, keying, and voice-overs.

Schedule Automation

To deliver all the capabilities described for this software A/V mixing process in the context of a real world broadcast operation, tight integration with a schedule automation process was essential. This meant allowing all features of the A/V process to be invoked directly from a prepared playlist with frame-accurate results. It also meant retaining the ability for realtime creative control and manual override from the operator's control position.

For example, in a traditional broadcast transmission chain, graphics for logos or CG overlays are stored on separate hardware components and triggered at the correct time by the automation system. Using the software-based approach, these graphics are simply another type of content file, and are held in a common "content store," along with video clips, audio clips, stills, etc. Thus graphics can be previewed, edited, sized, and positioned via the same operator interface as video clips.

Complex event sequences, such as "squeeze and tease" DVE moves combined with still backgrounds, graphic overlays and voice-overs, are increasingly common as a branding technique used by TV stations and networks. The content included in these sequences often changes with each use, making setting up and executing such sequences one of the most challenging parts of the scheduling and transmission process, especially if some steps require synchronized actions by multiple devices and/or operator intervention. Mistakes and resultant on-air errors are common.

Having the ability to schedule, preview, control, and render all elements of a complex sequence within one integrated software process controlled from one operator interface, makes management of these sequences much simpler and far less error-prone (**Fig. 4**).

The design of an operator control interface to transmission automation should be adapted to the requirements of each broadcast operation. Depending on the number of channels under automation and the type of content being transmitted

Figure 4. Single-channel control interface.

(e.g., movie channel, live sports channel, local network affiliate, "news wheel" channel, etc.), different combinations of single or multichannel displays including event lists, schedule timelines, and manual override controls may be appropriate.

The operator interface developed for Project 9, as shown above, provides both a graphical timeline and a detailed list to display the schedule of events for one channel. It also includes a variety of operator controls for channel selection, content selection, event editing, live logo insertion, schedule selection, and event triggering. In a multichannel operation, this might be the type of control interface used for a channel that required some operator intervention, such as manually triggered events (e.g., a commercial break following a live segment), live logo insertion, and schedule editing.

Technology already available from an earlier product development was enhanced to meet the full functionality objectives of Project 9, providing customized control interfaces for different operating environments.

Experience in beta test sites for a product based on Project 9 has shown that this attention to the design of the operator interface, and the tight integration between the schedule automation and the A/V processes, have produced a system requiring significantly less operator training time than that required for a traditional broadcast transmission chain with automation. It is expected that these same factors should also result in a lower error rate in on-air use.

Experience to date, suggests that the overall deployment time for this system (delivery to on-air) is less than that required for a traditional playout automation solution.

The Benefits of a Software-Based Approach

One of the largest barriers to entry as a content broadcaster is the cost and operational complexity of a hardware-based automated transmission chain. The costs for this traditional model are not just capital costs for equipment, but also the ongoing costs of managing, operating, maintaining, and updating a mixed bag of specialized hardware and software components.

Moving to a software-based solution means collapsing all the specialized, discrete components—both hardware and software—into a single, integrated software system operating on a standard IT platform. Stated differently, a software-based transmission system becomes an extension of an organization's existing IT infrastructure. Project 9 has shown that it is not only possible to originate high-quality video content from a standard IT server, but it can also be done with a level of quality and performance that meets the requirements of professional, commercial broadcast delivery.

For broadcast operations, the move to a software-based transmission solution on an IT platform brings additional benefits that are much more difficult to achieve with specialized broadcast hardware and conventional automation. Foremost among these are scalability and redundancy. These are broad topics and each deserves a detailed description. However, the underlying reason for the enhanced scalability and redundancy of this new model is simple: IT compared to broadcasting is a much larger business segment. Market size drives innovation and the availability of multiple solution options. In an IT-based broadcast system, this means a wide variety of options for content storage capacity, server and network expansion, and redundancy of all components.

An equally important consideration for commercial broadcasting is reliability. As a longtime provider of broadcast automation systems, experience has shown that, by far, the most troublesome part of a traditional automated broadcast transmission chain is the connectivity and control of multiple broadcast devices by the automation system. By eliminating both the physical connectivity and the proprietary protocol-based control of these multiple devices, the integrated software-based solution removes these two major causes of transmission chain failure.

Perhaps the least predictable but most certain realities of broadcasting today are continued change and the emergence of new requirements. As the market drives the expansion of content distribution paths from over-the-air, cable, and direct broadcast satellite to include internet and mobile devices, the channel formats that will provide the best return for each path are yet to be determined. Crafting new channels with the right mix of program segment lengths, effective branding, and commercial insertion for each distribution path will evolve through trial and error. To survive and succeed through this process, broadcasters will require tools with an unprecedented level of flexibility. The specialized hardware selected today may be useless for delivering content via tomorrow's new channels. The only sure solution to this dilemma is software that can adapt.

Conclusion

With the financial and competitive demands on content providers, the case for rationalizing the complexity of the broadcast transmission process is now too compelling to ignore. The continuing pressure to launch new channels—each one at lower cost—while retaining the production quality necessary for success in a crowded market calls for an alternative to the traditional broadcast model.

The opportunity to replace a complex, inflexible, hardware-based broadcast system with a more-capable, flexible, and scalable software-based solution at significantly less cost provides that alternative. The fact that this can be done as an extension of a company's IT infrastructure opens the door to a range of additional efficiencies as the traditional separation between business and operations "silos" in broadcasting disappears. For organizations new to broadcasting, this approach avoids developing these silos as a side effect of deploying a traditional broadcast model.

The development of Project 9 has produced a new paradigm for broadcast transmission. The rate at which this model will be widely adopted is as yet unknown, but initial experience is very encouraging. The beta version of a software product based on Project 9 has been demonstrated and deployed to a wide range of broadcasters, including individual TV stations, broadcast networks, large multichannel facilities, Telcos, and internet streaming providers. Among this group, recognition of the compelling benefits and demonstrated performance of the software-based model has been unanimous.

Presented at the 2007 SMPTE & VSF VidTrans Joint Conference in Orlando, FL, January 21-24, 2007.

The Authors

Ian Fletcher is chief technology officer at OmniBus Systems, where he oversees the development of a number of generations of automation technology, including OmniBus' innovative modular G3 software architecture and the groundbreaking iTX. This latest addition to the company's product range is a next-generation, software-based production and transmission solution for SD & HD playout that is revolutionizing the broadcast industry. Fletcher retains a key strategic role within the company as a key board member and is frequently asked to speak and write articles on the future of broadcast technology.

John Wadle began his professional career in software development and database design with a technology research company, BBN in Cambridge, MA. In 1975, he founded Federal Systems Associates in Washington, D.C., specializing in development of distributed database systems for U.S. government agencies. In 1989, Wadle founded ICA Systems Group as one of the pioneers in broadcast automation systems. With the acquisition of ICA by Columbine JDS in 1995, he served as VP of software engineering. Following the merger that formed Encoda Systems, he was named VP of product management at Encoda. In 2003, Wadle joined OmniBus Systems as VP of technology. He also serves as product manager for OmniBus' OPUS Content Management solutions.

Wadle holds a BA in mathematics from Northeastern University in Boston. He is a member of SMPTE and the Association for Computing Machinery (ACM).