Dialogic®
Making Innovation Thrive™

White Paper

IVVR Technology Introduction with Applications Using Dialogic® PowerMedia™ Host Media Processing Software

IVVR Technology Introduction with Applications Using Dialogic® PowerMedia™ Host Media Processing Software

White Paper

Executive Summary

As customer demand for video increases in today's mobile devices and conferencing applications, Interactive Voice and Video Response (IVVR) is becoming more prevalent. IVVR technology, an evolution of Interactive Voice Response (IVR), adds video picture to the IVR systems that many people have been using for years, thereby enhancing the voice calling experience.

This white paper briefly introduces IVVR technology, including how it is used and background in video technology and 3G-324M. Also presented are the significant features and issues of cellular video that are applicable to IVVR technology in today's world, such as video quality, bit rate control, fast call setup, and others. Also discussed, for developers seeking to add IVVR code to their Dialogic® PowerMedia™ Host Media Processing Software applications, are concepts such as the IP and 3G-324M sides of the PowerMedia HMP application for video, the Dialogic® Device Management API, stream and text overlays, video conferencing/video mixing/PIP, testing, and performance.

Note: Dialogic® Host Media Processing Software has joined the Dialogic® PowerMedia™ Media Processing Product Family and is now known as Dialogic® PowerMedia™ Host Media Processing Software (PowerMedia HMP).

IVVR Technology Introduction with Applications Using Dialogic® PowerMedia™ Host Media Processing Software

White Paper

Table of Contents

IVVR Technology Introduction with Applications Using Dialogic® PowerMedia™ Host Media Processing Software

White Paper

Introduction

This white paper provides a brief introduction to video technology, and background in Interactive Voice and Video Response (IVVR). Of particular note are the features and issues of video calling that pertain to IVVR technology — such as which codecs can be used; issues with voice quality, bit rate control, screen resolutions and frame rates; improvements in fast call setup; interactivity using DTMF; sources of video and audio streams (play/record), video transcoding, and signaling (SS7); hardware used (E1 or ISDN PRI, and others); and handset compatibility issues.

Also discussed are concepts for developers who are seeking to add IVVR code to their Dialogic® PowerMedia™ Host Media Processing Software (PowerMedia HMP) applications, such as the IP and 3G-324 sides of the PowerMedia HMP application, device connections using the device management API as a way to interconnect streams of audio and video from one side of the video call to the other, stream overlays that include image or text that can be applied on top of the video stream, video conferencing, testing, and performance.

What is IVVR and How Is It Used?

Interactive Voice and Video Response (IVVR) adds a video picture to the IVR systems that are already installed and familiar to many people. In this world of cellular 3G, handsets are usually equipped with cameras and color screens, and many such devices have the ability to do video calling; that is, adding a full duplex video picture to a voice call. IVVR technology can improve the video picture experience for users and enable them to get to the information they need more quickly than using just a recorded voice. In the following examples, IVVR could:

- Display a screen of options so the user can quickly select the desired option instead of listening to a long spoken list of options.
- Display a picture that conveys information faster than an audio description; for example, an online check-in system that displays the actual plane seat layout and selected seat (see http://www.youtube.com/watch?v=1MOhmjDQMs4 for a demo) [Speechstorm].
- Use animation or avatars to convey information.
- Stream live video (for example, in a traffic news system, a live stream shown from a street traffic camera).
- Be used to overlay information onto live video, as in a sports news application, where it may be desirable to show a “ticker tape” of results over a video stream.

Many of these applications can be seen in action at the Dialogic® YouTube site at http://www.youtube.com/user/dialogiccorp.

In many ways, IVVR applications can perform the same function as a dedicated application that may be found on an Android, BlackBerry, or iPhone handset. However, the following notable differences make video calling (or 3G-324M) an attractive option:

- Is an International standard built into many handsets. The protocols are standardized and the user does not need to load an app or configure the handset.
- Is not limited to smartphones, so it potentially can reach many more users. It is commonly recognized that about 85% of deployed handsets in the world today are feature phones, rather than smartphones.
- Does not need a data plan. It is estimated that worldwide only about a quarter of all cell phone users currently have a data plan.

As of 2010, video calling using IVVR technology has reached greatest acceptance in Asia, but deployments are going on in Europe, and now starting in North America (for example, Bell Mobility).

Introduction to Video Technology — 3G-324M Background

The 3G-324M specification was in use well before mobile devices started to have powerful CPUs and internet access, and also before phones and PDAs started to merge together into one category. When 3G-324M was being developed, it was decided that 3G-324M would use much of the existing cellular infrastructure; for example, the traditional 64 kbps voice channel would be used to carry video as well. This is possible because techniques for coding and decoding voice and video (codecs) had advanced so much in the 30 years since the launch of digital phone networks, that voice and video and some control data could all be packed into the one channel.

The 3G-324M protocol consists of the following:

- Control information (described by standards H.223, H.245, and H.324)
- A video stream, encoded via video codec; for example, H.263
- An audio stream, normally encoded with AMR-Narrowband (AMR-NB)

The video stream is allowed about 40 kbps of the bandwidth; the audio stream gets around 12 kbps, and a small amount of bandwidth is needed for the control traffic and framing overhead. It is significant that the video stays within its fair allocation, and this topic is discussed in the “Bit Rate Control” section.

In a video call, the phone establishes a 64 kbps connection, and then this pipe, instead of using only voice, carries the multiplexed 3G-324M session, which in turn carries voice and video.

From a quality perspective, AMR-NB, used for audio streaming, compares favorably to the legacy G.711 codec. The video stream is much different from those seen on TV or computer because the number of pixels in a frame is very small (matching the small size of cell phone screens). Also, the frame rate is reduced, since it has to work within a very tight bandwidth budget. The power of the cell phone CPU used to be an important variable, although this has become less of a factor over time.

Notable Features and Issues for IVVR

The following are some features and issues of cellular video that are of note from the point-of-view of IVVR applications.

Codecs

When the 3G-324M specification was first defined, the choice of video codecs included:

- **MPEG-4** — Perceived as the higher quality codec. Is also used in the .3gp files that are often downloaded to phones for video and/or audio
- **H.263** — Specified as the mandatory codec in the standard

Nowadays, with far more processing power to handle transcoding than could have been imagined at the turn of the century, video often uses the H.264 codec. H.264 has excellent quality (the products in which the H.264 family of standards can be found range from cell phones to High Definition TV), even in bandwidth-constrained applications like those used in cell phones, and demand is increasing for H.264 to replace H.263. The 3G-324M standard’s flexibility enables it to offer a list of supported codecs when each end of the connection specifies its needs (using the Terminal Capability Set [TCS]). This in turn enables the two parties to agree on a common codec, similar to the INVITE procedure used in the SIP protocol.

Also, although the 3G-324M protocol has a procedure for negotiating different audio codecs, in practice AMR-NB is nearly always used in deployed systems today. Some Cellcos are trialing (or starting to deploy) wideband (commonly known as HD Voice), so the wideband version of AMR, AMR-WB, is becoming a more common requirement for handsets and gateways.

Video Quality

Those familiar with digital TV know some of the things that can go wrong with a digital picture. Some of the notable issues include visible artifacts, “smearing” on fast moving images, and noticeable blockiness or missing color information when a picture changes rapidly. Getting a good quality video picture has many aspects to it, which will be discussed in the following sections.

Bit Rate Control

As mentioned previously, it is important that the video signal stays within the allocated bandwidth budget. When the bandwidth is exceeded, it can cause problems like lost frames (which disrupt the picture), or with the audio track slipping out of synchrony with the picture (for example, when an actor’s lips do not match the sounds being spoken). PowerMedia HMP makes use of sophisticated, Dialogic patent pending bit rate algorithm [Media] that gives it industry benchmark quality.

Resolutions and Frame Rates

A mobile device has a small screen, and so can tolerate a picture made of fewer pixels. The common pixel settings are QCIF and SQCIF (representing 176x144 and 128x96 pixels, respectively), which are quite small compared to a modern TV that might have a resolution of 1280x720 pixels. In some cases, gateways are required to “trans-size,” or convert from one size format to another when needed. For example, when connecting up a QCIF cell phone user to an enterprise video-conferencing system; the video has to be reduced significantly to provide the proper reduced resolution to the mobile device.

Table 1 shows the approximate data rates for popular video formats.

Encode Type	H.263 QCIF	HD 720p	HD 1080p
Resolution	175x144 @ 15 fps	1280x720 @ 60 fps	1920x1080 @ 60 fps
Bit Rate	64 kbps	20 Mbps	50 Mbps

Table 1. Approximate Data Rates for Popular Video Formats [ITU-T]

A television will have a frame rate of around 30 frames per second (fps) (or even 60 fps for a modern progressive scan TV). A mobile device network is not built to support high rates of update, and mobile video is generally geared to around 6 to 10 fps. Also, mechanisms are built into the codec to optimize the use of bandwidth by not sending all the available information every time. Within the stream, the whole picture (intra-frame or I-frame) is sent perhaps once every five seconds. In between, P or B frames are sent, and these are a kind of “delta” or difference frame either based on the previous frame or based upon the next frame. When the picture has static content (like a “talking head” on a news bulletin), the P or B frames can contain relatively little information, and yet the picture quality is good. Rapid movements, however, cause larger volumes of data in the delta frames.

If an I-frame is lost, then the structure of the picture can be lost. The picture’s resulting blockiness or ghosting, or strange color effects occur because the remaining frames base their changes on the last frame, which, in this case, is missing. In a cellular environment, it is good practice to send I-frames every five seconds, so that if there is a drop-out, the picture can be restored within that time.

One of the features of video gateways (and technology like PowerMedia HMP) is the ability to transrate (that is, change the frame rate when needed), and this must be done efficiently and with good bit rate control.

Fast Call Setup

A defect of early 3G-324M systems was that it would take a relatively long time (on average, 6 to 8 seconds) to establish a video picture at the start of a call. This was due to the amount of time needed to negotiate between handset and gateway. Optimizations made to the procedures, namely to Media Oriented Negotiation Acceleration (MONA) and Windowed Numbered Simple Retransmission Protocol (WNSRP), greatly reduce the delay. PowerMedia HMP supports these procedures, although not all handset vendors consider them mandatory.

Interactivity Using DTMF

For IVVR, it is necessary to have two-way communication so that the user can control the application. Control is accomplished with Dual Tone Multi-Frequency (DTMF) digits, in the same way as for a conventional IVR (actually, H.245 is used instead of audible tones, but the effect is the same to the central application). IVVRs are typically controlled either by voice (speech recognition) or via keypad digits 0-9, *, and #.

Sources of Video

An IVVR can get its video (and audio) stream from a variety of sources. The following are some source possibilities:

- **Video files (play/record)** — IVVR systems will likely play audio, and many want to play from stored video files. Sometimes these files will be standardized, like the ".3gp" file, and in some cases raw streams will be convenient (PowerMedia HMP supports both 3gp files and a Dialogic proprietary format for quick streaming delivery).
- **Video servers** — Sometimes external servers contain the video, and RTSP will be used to control it. These "trick play" or DVR features allow a video server to pause, or the user to navigate back or forth to a new point.
- **Static image files** — Static information (like menu choices or the seat map of an airplane) could come from a static file; for example, in JPEG or YUV format. It is useful to have the ability to "play" a static file, generating a video stream to the handset. Sometimes content (like a ticker tape or logo) needs to be overlaid on top of live video in real-time.
- **Hairpinning** — In some cases, two video call legs are bridged (also called "hairpinned") together.

Transcoding

Video streaming used in IP networks is often different from that used in a 3G network. In these cases, video transcoding may be required. In IP, video can be streamed in H.264, MPEG-4, or H.263 codecs, via RTP streams, and normally the resolutions and bit rates of the IP network are higher than the 3G side. It will be the job of the video gateway — in this case, a combination of PowerMedia HMP and a Dialogic® HMP Interface Board (DNI Board) — to transcode from one of these IP protocol profiles as needed to, for example, H.263, the mandatory codec on the 3G side. PowerMedia HMP has this ability, and may also at the same time be transsizing the picture and transrating to reduce the frame rate.

Another major reason to use a video gateway for video transcoding is to be able to generate I-frames on demand to the mobile device. Often, in the mobile environment, poor cell reception or other factors can cause errors in the video data. Because of the nature of video compression, these errors will build over time as each delta frame builds on the previous errors. Although video transcoding can be expensive in terms of the CPU, the ability to generate I-frames on demand allows quicker recovery of the video stream at the mobile device and can significantly increase customer satisfaction.

Signaling

A variety of signaling protocols are still used in Telco networks. In small-scale systems, ISDN or Q.931 signaling is used, but Telcos often prefer native SS7 support, as this is how they interconnect their boxes in the backbone. Traditionally, signaling was done using the ISUP protocol, which has many international and national variants. Increasingly, Telcos also use BICC, for example in the case where NbUP is used to transport the video media. PowerMedia HMP supports these options, and the Telcos determine the proper configurations.

Hardware

3G-324M is a PSTN-protocol, so it requires PSTN hardware termination, for example E1 or ISDN PRI. In the PowerMedia HMP architecture, the physical link is generally a DNI Board, with PowerMedia HMP providing the signaling and media intelligence.

On the signaling side, large scale systems use ISUP or BICC to control large numbers of channels. Physically, a Dialogic® SPCI or DSI SS7HD Network Interface Board provides this connectivity, again with PowerMedia HMP providing the overall intelligence (the Dialogic® Global Call API library hides the topology of the SS7 signaling from the application, so that Dialogic® DSI SS7 Boards or the Dialogic® DSI Signaling Interface Unit platform can be used together with PowerMedia HMP).

Handset Compatibility Issues

3G-324M-capable handsets have a high degree of conformance. Some details are available in the *Dialogic 3G-324M API Programming Guide and Library Reference* (see "For More Information"). There could be variances depending on the required codec support and implementation of Fast Setup. Also, Cellcos often have their own mandatory requirements documents for handset and gateway vendors.

PowerMedia HMP and DNI Boards have been verified against a multitude of mobile devices. Dialogic also is a regular attendee at International Multimedia Telecommunications Consortium (IMTC) 3G-324M Interop events, where advanced features are tested, like H.264, AMR-WB, and MONA, that may have been recently incorporated into mobile vendor products.

Adding Video to Dialogic® PowerMedia™ Host Media Processing Software Applications

In a common scenario, two sides to the PowerMedia HMP application for video exist: an IP side and a 3G-324M side. (Note that it is also possible to use PowerMedia HMP as the media server in conjunction with the Dialogic® Vision™ 1000 Video Gateway as the conversion gateway.) This allows a video back-end (perhaps streaming H.264 or MPEG-4 video) to be interconnected with a 3G-324M video call. Developers who have used PowerMedia HMP for VoIP applications previously will notice that these are familiar API calls. A notable difference when using PowerMedia HMP as a media server and using an external video gateway (Vision 1000 Video Gateway) as the physical interface (instead of using PowerMedia HMP with a DNI Board) is that the 3G-324M part of the call is provided by the m3g device and its associated API.

The IP Side

On the IP side, a SIP video connection typically will have at least two RTP streams, one for carrying the audio and one for the video. The SIP INVITE procedure allows the calling party to negotiate audio and video codecs with the far-end (media server and others). The ipm device manages the RTP part of the call leg, and ipm_StartMedia is used to specify the parameters of the video session. The dx device is used for audio processing, and the mm device provides the multimedia functionality. The main job of the mm device is to do multimedia play and record, but it also has other capabilities, such as DVR control, still image play, and still image capture. The dx device is still used in many scenarios for in-band DTMF detection and generation. If an application uses RFC2833 or SIP Info messages for DTMF control, then a dx device may not be used at all. In-band or RFC2833 DTMF generation to the network requires a dx device.

Call Control

The IP side of a PowerMedia HMP application manages the SIP call control and media in the same way that it does for a VoIP SIP audio call. SIP call control is licensed with an IP call control resource, which is handled by the Dialogic® Global Call API and Dialogic® R4 API. Within the SIP messages (INVITE, ACK, and so on), the supported media codecs are described by the SDP (Session Description Protocol) within those messages. The supported audio and video codecs are listed in priority order.

When moving from a voice-only application to a video application, note that PowerMedia HMP video applications must be written in 3PCC (Third Party Call Control) mode. This differs from many PowerMedia HMP "audio only" applications that were written in Global Call 1PCC (First Party Call Control) mode. In 1PCC mode, the Global Call API/R4 API logic figures out for the application which codecs to start and then issues the appropriate ipm media function calls on behalf of the application. For a video application, 3PCC is the mode in which an application uses the Global Call API/R4 API to negotiate the SIP session, but then the application must parse the SDP and start the RTP media by calling the appropriate ipm library commands.

The move to 3PCC adds codec negotiation work to the application, but provides more granular control over media handling. To facilitate SDP parsing, the PowerMedia HMP code comes with an open source SDP library in the ~/dialogic/demos directory. PowerMedia HMP video application demos use this SDP library code to do SDP parsing. Moreover, for those moving applications from 1PCC to 3PCC mode, a Dialogic online training module called, "Adding Video to a Dialogic® Global Call /R4 IVR Application" is available and covers the steps involved in this conversion (see "For More Information").

Media: IPM for RTP Streaming

After the application parses the SIP SDP parameters and negotiates the media codecs, the ipm library is used to start the RTP streaming. One ipm device supports both audio and video streaming, so starting a multimedia (audio and video) stream is the same as starting an audio-only RTP stream, with the addition of video codec information on top of audio codec information in the ipm_StartMedia function. In the ipm_StartMedia function, an application would provide the local and remote video RTP and RTCP ports, and the video codec specifics, such as video codec type, RTP payload type, profile and level, resolution size, bit rate, and frame rate. For codecs such as MPEG-4 and H.264, decoder configuration information is also provided.

MM for Play and Record

The source of multimedia within PowerMedia HMP is the mm device, which supports playing proprietary PowerMedia HMP .vid/.aud files, as well as standard .3gp files with supported codecs. The PowerMedia HMP proprietary files are optimized for streaming and low latency time to first packet because they are files with pre-packetized RTP media in native format. The supported proprietary codecs for audio are G.711, G.723, G.729, G.726, AMR-NB, AMR-WB, and G.722; and for video are H.263, H.263+, MPEG-4, and H.264. Users choosing to use the proprietary formatted files for native play and record capability can cut down significantly on CPU cycles. The PowerMedia HMP proprietary file format also contains a synchronization offset between audio and video for audiovisual synchronization purposes. This is significant for files that are recorded and synch points that are used for DVR controls. The mm device also supports PCM, PCM 16k linear, and WAV (8k or 16k linear).

The mm device uses simple functions mm_Play and mm_Record for play and record, respectively. If the mm play function contains both audio and video codecs, then the play will remain synched; otherwise, a separate mm_play can be issued once for audio and again for video. This may be useful, for example, if the user wants to play a video file with separate audio tracks that may be shorter in duration. The mm device also supports simultaneous play and record on a single mm device.

The mm device play also supports playing a still image. To play a still image, a user calls mm_Play with a JPEG or YUV file. Internally, the image is sent through the video encoder and sent out as an I-frame plus subsequent P-Frames matching the video frame rate and bit rate set for the session. In this way, up to approximately ten separate images can be streamed per second, making it useful for users looking to create movement from several pictures (for example, being able to show the specific hands being played in a card game). The play "still image" feature is available only if a transcoder is present in the video connection.

The mm device can also capture a still image from a video stream. The mm_capture API is used to capture a moment in a video stream being played by the mm device to a YUV image. A user pauses the stream with mm_pause, captures the image with mm_capture, and resumes with mm_resume.

DVR controls are supported by the mm device API. The DVR mechanism uses the mm_pause, mm_seek, and mm_resume APIs to stop the current stream, jump to a new position in the file, and resume playing from that location. DVR controls are available on PowerMedia HMP proprietary files only but are not supported for PCM or wav audio files. Video files must have synch points (which represent I-frames) to allow skipping forward or back and starting the stream with a reference frame. Some peculiarities can occur when trying to "seek" in a video file because the video must start on an I-frame, and may need to skip to the nearest I-frame, which may actually end up causing the video to skip backward in time. These are common DVR scenarios, but are notable, especially when creating video files.

The 3G-324M Side

The Dialogic® 3G-324M API uses a device called m3g. Dialogic m3g documentation, *Dialogic 3G-324M API Programming Guide and Library Reference* (see "For More Information"), describes how to establish calls as previously discussed, covering selection of codecs, frame rates, and picture size.

The m3g device is an intermediate component that is neither a media source nor a media sink. The m3g device completes the 3G-324M peer-to-peer protocol and terminates the protocol session. On one side of the m3g device is the transport of the multiplexed bitstream, on the other side are the demultiplexed audio and video streams. Therefore, the m3g must be connected internally on both sides to other PowerMedia HMP devices. On the transport side, the m3g device is connected to either a dti device (i.e., a DNI Board) for connection to the PSTN network or to an ipm device for connection to an NbUP IP network. On the media side, the audio and video streams can be routed to an appropriate audio or multimedia device.

The audio and video connection of the m3g device determines the function that will be provided to the remote 3G endpoint. For example, a common scenario is to connect the m3g device to an ipm device for a 3G–to-IP gateway function. As discussed in "The IP Side" section of this white paper, the IP side can be a media server, RTSP server, or IP endpoint.

The m3g device can also be connected directly to an mm device for direct play/record to/from a 3G endpoint. A benefit in this case may be lower latency and the ability for the application to play an announcement without sending the RTP stream to another server. For example, the audio and video are recorded in a file (monitoring) while the call is being routed to an IP endpoint. Similarly, the m3g device can be connected as a party to an audio-only or multimedia conference. As with the direct connection to an mm device, this can reduce latency and provide one platform so that traditional PSTN calls can be conferenced with video calls.

Video Call Completion to Voice (VCCV) is a function that allows video calls to correctly connect, even if the recipient has only a voice channel available. For example, the call recipient may be temporarily out of range for 3G coverage, or perhaps the recipient only has a 2G handset. VCCV is beneficial for the caller (who otherwise would have to redial) using a voice-only call. Also, by encouraging users to make more video calls, VCCV can be useful for the Telco. In systems using PowerMedia HMP, VCCV can be achieved by connecting the m3g device audio to a dti device (DNI Board) timeslot. This allows the audio path to be connected end-to-end, even though only one party has video.

Finally, the audio and video from one m3g device can be routed directly to another m3g device to facilitate 3G-to-3G bridging, outward dialing, or Multimedia Ring Back Tone (MRBT) applications. In addition to Color Ring Back Tone (CRBT), which allows the user to customize the ring that callers will hear, video can be used, providing the caller with video/audio during the ringing phase.

Device Connections

The device management API (dev) functions, like dev_Connect and dev_PortConnect, provide a way to interconnect streams of audio and video from one side of the video call to the other. As the Dialogic® Device Management API evolved from an audio-only API to multimedia, the standard connection mechanism to connect internal devices has also evolved. In traditional voice applications, a 64 kbps timeslot of the CT Bus was used and each device had an audio transmit timeslot that other devices listen to. For multimedia, the Device Management API has evolved to become an internal port-based connection method, similar to RTP ports, where the source device transmits audio and video from its internal transmitter (tx) ports to one or more other devices' internal receiver (rx) ports.

One benefit of the Device Management API is that audio and video can be forked from one tx port to multiple rx ports. An example of this would be in an announcement server where the same multimedia file is played to many different endpoints. Another example is the ability to send an audio or video stream to an endpoint while also forking the same stream to an mm device to capture a recording.

It is significant to point out that any port connection between devices (audio or video) can be transcoded or native. In order to turn on transcoding between two devices, a transcode flag is set in the tx to rx connection. This enables the PowerMedia HMP firmware to convert from the format of the tx device to the format of the rx device. The same flag could be set to native to turn off transcoding.

A native connection means that the audio or video data remains in its compressed format from source to destination device. A transcoded audio connection means that the audio will be converted into a common uncompressed linear PCM format before being compressed in the destination format. (This conversion to linear PCM is what happened in the traditional timeslot-based model.) A transcoded video connection means that the video will be converted to an uncompressed YUV format before being converted to the destination format.

Stream Overlay/Image and Text Overlay

Stream overlays include image or text that can be applied on top of the video stream. Image overlays and text overlays can be added only on a video-enabled device (ipm, mm, m3g, or cnf/mcx party) that has video-transcoded video connections. When the connection is transcoded, the PowerMedia HMP firmware uncompresses the video data into a common YUV format before compressing the data again in the remote video format. YUV format is similar to an RGB format in that the PowerMedia HMP firmware has access to each pixel in the frame and keeps a running uncompressed snapshot of the video at the current moment in time. It is in the YUV domain that the pixels can be altered and an overlay can be applied to the video stream.

Overlays are created and applied with the Dialogic® Media Toolkit API. The application note, *Using Dialogic® Media Toolkit API for Image Overlay on Video Streams* (see "For More Information"), covers how the Media Toolkit (MTK) API can be used by an application so that a user can play an image (YUV or JPEG) over streaming video. Text overlay (targeted to be a feature in PowerMedia HMP in the second half of 2010) will support font sizes and colors supported by the OS, along with left, right, up, or down scrolling. At the time of this writing, a text overlay can be applied as an image, by using a tool like ImageMagick to convert text into an image before applying the image overlay on the PowerMedia HMP device. The application note, *Using Dialogic® Media Toolkit API and ImageMagick for Image Overlay on Video Streams* (see "For More Information"), also covers using the MTK API with ImageMagick for text overlay.

As of this writing, PowerMedia HMP supports up to two overlays per device, but the overlays can be applied at the device in the direction of the network or internal device connection. Therefore, between two devices connected by transcoding, up to four overlays can be applied in one particular video stream direction. For example, if an ipm device is connected to an mm device with video transcoding enabled in the mm-to-ipm direction, then two overlays can be applied on the ipm device handle in the network direction, and two overlays can be applied at the mm device in the device direction, providing four overlays on the video stream transmitted from mm to the IP endpoint.

The MTK API consists of general mtk_ API calls to create the image template, overlay builder API (ob_) to build an overlay image, and stream manipulation API (sm_) to apply overlays to a stream. An application designs the image template by providing the media information (such as YUV or JPEG), creating a bounding frame, including bounding frame size, pixel location, and justification of the image in the frame, and then binds the frame to the template to create the overlay. The overlay can be applied to a video enabled device by calling sm_ApplyOverlay, using the PowerMedia HMP firmware device handle and the handle of the overlay.

Video Conferencing/Video Mixing/PIP

PowerMedia HMP supports video conferencing, customizable Picture In Picture (PIP), and video stream-on-stream overlays by using the video conferencing (mcx) resource. The mcx device is a video-enabled version of the cnf audio conferencing device.

In this white paper, the terms video conference and video conference device are used to describe the multimedia mixer mcx device. Note, that as a multimedia mixing component, the mcx device can do much more than just video conferencing.

A video-enabled endpoint (ipm, mm, or m3g) can be connected as a party in a video conference. Additionally, the mcx conference device also supports connection with dx (voice) and dti (DNI Board) devices for audio-only support in a multimedia conference. The devices connected to the video conference can be any of the supported video or audio codecs that can be used with transcoding. Thus, video can be H.263, H.264, or MPEG-4; and audio can be a supported narrowband codec, such as G.711a/u, G.729, G.723, AMR-NB, G.726, as well as the wideband codecs G.722 and AMR-WB.

The video conference component provides one mixed (the audio of all parties is mixed, and typically the video stream is selected from one of the parties, for example, the loudest talker) conference layout to the devices that receive the conference output. The conference layout can be thought of as the view of mixed video streams provided to the viewer. Each member of the conference gets the data returned in the same codecs, resolution size, and bit rate as provided to the conference. An application may be selected to provide the endpoint with the conference output or the endpoint may be connected to another device source. For example, during a video conference, the speaking member may not want to see himself/herself when he/she is talking in a single party layout. In this case, the source to the speaker's endpoint instead could be an mm device, providing teleprompter notes or something less distracting as the person speaks.

The video conference layout is designed with the MTK API. One of the functions used to create a conference layout is the layout builder (lb_). The layout builder API is used in conjunction with the cnf conferencing API to apply the conference layout to a particular mcx conference device. The video conference layout is drawn by sizing up several video regions to fill the viewable area. The regions are designed using functions such as:

- **lb_Set Rect** — To set the region dimensions and placement
- **lb_SetPriority** — To set the overlap priority of the region
- **lb_SetDisplayMode** — To set whether the region will be blank, frozen or live streaming video

Additionally, the region can be set to active-talker (for example, the loudest speaking people in a conference could be used to automatically select which video streams get shown in the conference video frame) mode with lb_GetSelectionMode in case the application wants any region to be active-talker switched based on the PowerMedia HMP firmware active-talker list. Any video party can then be placed in any region, with any number of regions capable of being created per layout (there is a practical limit to the number of regions in a CIF or QCIF resolution screen that are visually useful). The layout builder API has a few predefined layouts, including a single-region layout, as well as four-region, six-region, and nine-region layouts, each with the possibility of active-talker control for specified regions. However, these are only example templates; the real power of the video conferencing component lies in the ability to draw custom regions and custom layouts.

The video conferencing device supports custom video layouts, which is useful in video conferencing and PIP applications. Since the video conference and party devices are port-connection based, the application can use the video conference device as a video mixer, audio mixer, or multimedia mixer. The application developer can design a video conference layout with a non-limited number of region designs that can start both inside and outside the viewable area. Additionally, once a video layout is created, it can be applied to the video conference at any point, with almost instantaneous switching from layout to layout by calling the cnf_SetVideoLayout API. In this way, it is even conceivable that the application can make the regions in a conference appear to move by calling these functions with slightly modified layout properties within a tight programming loop. Within a layout, sources to each video region can also be changed almost instantaneously as the video conference device can place any party in any region in the time it takes to apply the cnf_SetVisiblePartyList API.

Testing

The 3G-324M protocol essentially uses a clear 64 kbps channel to carry its information, and the 3G-324M part of the code is not affected by how that channel was connected. This means that with a bit of extra coding, it is possible to interconnect two calls (in a loopback), so that the PowerMedia HMP gateway is talking to itself. This is significant because not all testing has to be done in cellular network using real handsets; many functions (such as perhaps load testing) can be done in a controlled (and reproducible) environment.

Performance

Bear in mind that IVVRs will not scale to as large a number of users as equivalent IVRs. The processing of video in real-time can be expensive, so the actual number of concurrent calls will depend heavily on which operations an application needs, the resolution size, codec, and bit rate of the video sessions, and upon the power of the CPU(s) of the server. For example, an application that does full transcoding from MPEG-4 to H.263 has a heavier processing load than if the app is simply reducing the frame rate.

Dialogic's Global Call API/R4 API SDK includes a 3G-324M demo application with source code, and the Dialogic website has applications with source code (http://www.dialogic.com/products/ip_enabled/download/hmp-demos/default.htm). The SDK can be used as part of development and testing efforts, and it supports some advanced features like MONA and RTSP (DVR).

A Note on AMR-NB and AMR-WB

Using the AMR-NB and/or AMR-WB resource in connection with a Dialogic® product described herein does not grant the right to practice the standard(s). To seek a patent license agreement to practice the standard(s), contact the VoiceAge Corporation via http://www.voiceage.com/licensing.php.

References

[ITU-T] ITU-T Recommendation H.264: Advanced video coding for generic audiovisual services, Approved 03-2005.

[Media] Supplied by Dialogic® Media Labs http://www.dialogic.com/medialabs/default.htm.

[Speechstorm] Dialogic® demonstration of IVVR (Interactive Voice & Video Response), using a 3G mobile device to place a video call and use an airline check-in service developed by Speechstorm. See also http://www.speechstorm.com.

For More Information

Application Notes

Establishing a MONA 3G Multimedia Session with the 3G-324M API

Using Dialogic® Media Toolkit API for Image Overlay on Video Streams

Using Dialogic® Media Toolkit API and ImageMagick for Image Overlay on Video Streams

Dialogic Products

Dialogic® DSI Signaling Interface Unit

Dialogic® HMP Interface Boards (DNI Boards)

Dialogic® PowerMedia™ Host Media Processing Software

Dialogic® SPCI2S and SPCI4 SS7 Boards

Dialogic® Vision™ 1000 Video Gateway

Signaling and SS7 Components

Supporting Documentation

For the latest Dialogic® documentation online, see www.dialogic.com

Dialogic 3G-324M API Programming Guide and Library Reference

Dialogic 3G-324M Multimedia Gateway Demo Guide

Dialogic Device Management API Library Reference

Dialogic Global Call API Programming Guide

Dialogic Global Call IP Technology Guide

Dialogic Global Call ISDN Technology Guide

Dialogic Global Call SS7 Technology Guide

Dialogic® Host Media Processing Software Release 4.1LIN information

Dialogic IP Media Library API Programming Guide and Library Reference

Dialogic Multimedia API Programming Guide and Library Reference

Dialogic Multimedia Demo Guide

Other

Dialogic online training module, "Adding Video to a Dialogic® Global Call /R4 IVR Application."

Many of the IVVR applications are demonstrated on the Dialogic® YouTube site at http://www.youtube.com/user/dialogiccorp.

Training: A one day "School of Video" course run by Dialogic engineers is a hands-on course for developers including instruction and code-building lab sessions. Contact your local Dialogic representative for details on the schedule.

Dialogic
Making Innovation Thrive™

www.dialogic.com

Dialogic Corporation
9800 Cavendish Blvd., 5th floor
Montreal, Quebec
CANADA H4M 2V9

07/10 12026-01