Smooth Integration of Mobile Video Telephony to Windows Mobile Smartphones

by Tsahi Levent-Levi

[Editor's Note: Tsahi Levent-Levi works for RADVISION, which offers products for the 3G-324M protocol he discusses in this article.]

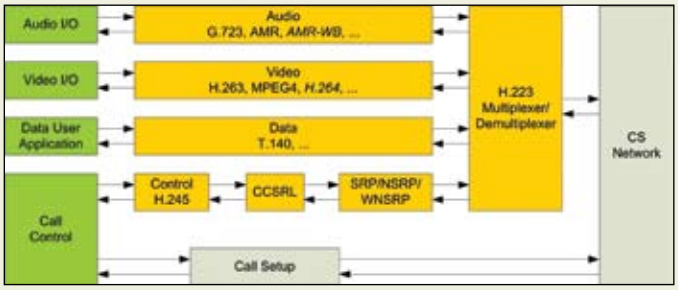
3G video telephony is usually implemented over the 3G-324M protocol. Today, handset manufacturers who wish to roll out 3G UMTS or TD-SCDMA handsets are well aware that 3G-324M support is a mandatory requirement for success. As Windows Mobile gains popularity and becomes more widely deployed in Smartphones and PDAs, the smooth integration of 3G-324M in Windows Mobile is becoming more and more important. How can this be accomplished?

Mobile video telephony

Today, the UMTS (WCDMA), and its Chinese equivalent, TD-SCDMA, enable a circuit-switched data connection of 64 kilobits per second. This connection is used for real-time video telephony over mobile networks, and is a popular service on UMTS networks in Japan, Korea, Taiwan, Europe, and many other countries and regions around the world. As operators are looking for additional services to leverage their networks and increase their income per subscriber, the addition of real-time video has become a real opportunity for growth. 3G handset manufacturers must include video telephony in their products. To do this they use the 3G-324M standard.

3G-324M overview

3G-324M is based on the ITU H.324 standard and is designed to send several different channels over low bit-rate networks. In 3G, this rate is 64 kilobits per second over a circuit-switched UMTS or TD-SCDMA connection.



The 3G-324M standard design

This technique for mobile video telephony is currently the best solution available for handling latency and roundtrip delay issues, implementing authentication and billing mechanisms, and minimizing equipment upgrade costs for operators.

Since we have very limited bandwidth for the large amount of data required (audio, video, and some control), the 3G-324M standard employs a multiplexer (H.223), which takes different channels and multiplexes them together into a single bitstream—while adding as little overhead as possible. This lets the actual audio and video exploit as many bits as possible.

The actual implementation of the 3G-324M standard is usually not done by the application developers, but is outsourced to other companies whose core competency is developing toolkits for communication standards and that offer a 3G-324M toolkit.

As a standard, the 3G-324M protocol does not need to know who is actually dialing and connecting the bitstream, or how the data is being sent and received. This makes it a superior solution for other scenarios as well, but for application developers, it introduces an additional complexity because they have more to implement.

There are three different issues application developers must deal with once a 3G-324M stack implementation is available:

1. Integration with the network through the baseband. This includes dialing and tearing down calls as well as sending and receiving 3G-324M related bitstreams using the H.223 component.
2. Integration with the media codecs. This includes audio and video codecs, including optimization when required. In order to achieve the best quality, auxiliary algorithms such as AEC (Acoustic Echo Canceller) and NR (Noise Reduction) are also required.
3. Building the application itself.

This article discusses these three issues, focusing on the integration with Windows Mobile 5.0 for Smartphone products.

Windows Mobile integration challenges

Properly integrating video telephony with the Windows Mobile platform involves more than simply writing an application. The Windows Mobile 5.0 Smartphone specification integrates all the mobile elements incorporated in the handset in order to ensure that users experience a unified look and feel that is intuitive and easy to learn and master. Customers expect video telephony to be an integral part of that look and feel and the functionality of the handset—and not just another third-party, standalone application that needs to be installed and maintained.

Windows Mobile comes equipped with interfaces to facilitate the integration of video telephony applications. Some are related to the network and the media codecs, but there are application-specific interfaces required as well.

Chipset architecture challenge

Usually, a handset using Windows Mobile will include two or more CPUs:

- **Application chip**: This chip is the one that actually runs Windows Mobile. It is used by the operating system, as well as the various applications and serves as the actual user interface for the end customer.
- **The baseband**: This chip is responsible for communication with the baseband by using RF. It usually uses a proprietary operating sys-

Tsahi Levent-Levi is a seasoned product manager and system architect in the telecom industry. His experience includes product management and development, project management, with a strong background in development and management of complex VoIP projects. Tsahi currently manages a wide range of products in RADVISION's line of developer toolkits and platforms, including the 3G-324M Toolkit for 3G networks and devices, which enjoys the largest market share in the industry. In addition, Tsahi serves as co-chairman of the IMTC 3G-324M Activity Group, which focuses on interoperability issues. He regularly participates in related ITU and 3GPP standardization conferences, where he works tirelessly to improve the standards for the industry as a whole, and for the 3G community specifically. He can be reached at tsahil@radvision.com.

tem, or a scaled down real-time operating system.

• **Hardware accelerator**: This optional chip handles the video and audio coding to reduce complexity and increase battery life. It can sometimes add 3D capabilities to the handset, which are not directly related to video telephony.

In these standard architectures, 3G-324M is usually optimally located on the application chip for several reasons:

1. The application chip runs Windows Mobile. Because the goal is to achieve the best user experience possible, maximum integration with the user's operating system, in this case Windows Mobile, is preferred.

2. Windows Mobile interacts with the baseband using a layer called TAPI and it is easier to handle this layer when 3G-324M is located on the application chip itself due to the architecture of TAPI.

3. Development and debugging on the baseband usually requires a lot of time and expertise. Development tools for these platforms are scarce and less advanced than for Windows Mobile. Since time-to-market is of prime importance, developing more on the application chip and less on the baseband is naturally preferable.

4. Developers have more control over the application chip's resources and peripherals than the baseband's. Any bells and whistles required can be implemented on the application chip, but might not be possible on the baseband chip.

Baseband and TAPI challenges

Because video telephony is a networking application, it should be connected to Windows Mobile through TAPI (Telephony API). Usually, a Windows Mobile handset will be based on hardware architecture with two or more CPUs. One is responsible for networking, and is known as the baseband. The other is responsible for the rest—in our case, the application chip running the Windows Mobile operating system itself. The actual interface between the application chip and the baseband is done using a RIL (Radio Interface Layer). This layer is not developed by Microsoft, but rather by the baseband provider or the integrator of the handset. Whatever the case, it is the realm of the party responsible for the interworking of the application chip, the baseband, and the operating system.

It is important to remember that 3G-324M communicates over a 64 kilobits-per-second circuit-switched connection. This is done over the network by sending a 160 bytes packet every 20 milliseconds. Do the math and you realize what a challenge it is to maintain quality. Overflowing the baseband or missing timeslots simply means degraded audio and video quality—an unacceptable option. For this purpose, great care should be taken when integrating with TAPI and when developing the RIL—issues relating to timing mechanisms and buffer management should be taken into account:

• Do we wish to send 160 bytes every 20 milliseconds, or is 320 bytes every 40 milliseconds acceptable in terms of delay?

• Can we use a "sleep" command or do we have to use internal baseband buffer management?

• Do we use the internal CPU clock, which may not be accurate, or try to use the baseband's clock for sending data?

The actual answers to these questions will directly affect the perceived video quality of the handset, so they must receive proper attention in the design phases of the integration.

Media codecs and Direct Show

When deploying video telephony, we would like to see both the audio and video channels opened and properly utilized—despite bandwidth limitations. Doing this requires codecs that compress and decompress the audio and video signals so they can be transmitted over 64 kbps, the gross bandwidth available. Effectively, the application is left with 60 kbps, or less, dedicated for both media types. This is quite limiting and presents two challenges:

• Optimization: A good video codec (MPEG4 or H.264) is CPU-intensive. By definition, it will take over the application chip's resources, leaving you with one of the following choices: optimize it for your CPU; use a more powerful CPU; or use hardware acceleration. My choice would be hardware acceleration. In addition to leaving more room for the CPU to breathe and allow other applications to run simultaneously and smoothly, it is also likely to extend the battery life of the handset—no easy feat when deploying video telephony.

• Direct Show: This issue is closely related to Windows Mobile. Everything needs to be properly integrated with Direct Show. If it isn't done correctly, end users will regret the day they purchased the phone every time there is an incoming video call while listening to their favorite MP3 songs. And that will be the least of your problems. Improper integration could lead to inferior media quality or a low frame rate—which may cause customers to run and buy a different handset from another vendor.

Remember that mobile video telephony is all about having video in the call. You should get the best platform with the best codecs and then integrate them in a manner that will keep your customers happy with what they see and hear.

Phone Canvas

Windows Mobile 5.0 supports an advanced level of customization in terms of the phone's look and feel. This allows differentiation between different manufacturers using this same operating system. To take full advantage of this component, developers should plan 3G-324M integration into the platform with these issues in mind. Correct integration requires adding configuration parameters to the phone settings and defaults; reusing configuration/data from voice-call settings; changing the menus; interfacing to hard-buttons; and sometimes getting events from other OS/phone modules.

Fully integrating the application will allow for unified branding by operators and an enhanced user experience.

Contacts and Call History

Using the same features for voice calls enhances the user experience. Users expect to have the same positive experience for video calls as they have for audio calls, and of course, enjoy their 3G handset more than their "old" 2G handset. This means using the Call History feature most mobile handsets have had for years, with lists of missed calls, dialed calls, incoming calls, dialing from an address book, etc.

Video calls will become one of the options on the default dialing screen.

• Contacts: Windows Mobile enables users to easily synchronize with their desktop Microsoft Outlook Contacts list, effectively creating a single address book for both the desktop and mobile handset. This powerful feature allows one-button dialing directly from the contact information. Users should expect no less for video calls. Easy dialing and a user-friendly interface for video calls will eliminate complexity and will encourage increased usage of video services.

• Call History—Call History is the component in Windows Mobile that logs Missed / Dialed / Incoming voice and video calls. An implementation of video telephony should take care of this aspect as well, enabling the user to dial video calls from these lists directly.

The integration of the application with these features must be done

Video calls will also become an option in Outlook's Contacts.

Users can initiate video calls from the Call History list.

If a video call fails, users can then initiate an audio call.

on the operating system level in order to have a single, unified look and feel for audio and video calls.

Interaction

A Windows Mobile Smartphone usually offers a wide range of applications to end users. The interaction between different applications regarding audio and video interfaces needs to be taken into consideration.

Before deploying a solution, testing the developed application should cover checking more than whether simple video call dialing and receiving scenarios work perfectly with deployed handsets and with existing servers, such as mail servers, media servers, etc. It should also cover scenarios to make sure that different phone applications do not cause problems. Examples of these are SMS messages received while in a call, incoming calls accepted while listening to an MP3 audio file, a call is being held while another application is being launched and then closed, etc. The possibilities are endless. In short, the more testing that's done, the better the end users' experience is going to be.

Mobile video telephony is coming

Windows Mobile is an advanced operating system that is quite suitable for the 3G arena. As we move towards total fixed and mobile convergence, mobile handsets are becoming more complex, with a wide range of applications integrated within. In order to take full advantage of Windows Mobile, an application as important as video telephony must be implemented with extreme care. The integration must take into consideration every aspect and requirement, as well as Windows Mobile architecture and interface. Exploiting the full power of Windows Mobile will guarantee a high quality product that delivers a superior user experience. Failing to do so will inevitably result in less appealing handsets for consumers. ■

Windows Mobile videophone announced

The Loox 830, a Windows Mobile phone with the videophone features discussed in this article, has been announced by RADVISION (who designed the advanced video application), and Compal Communications Inc (who will build the unit). It is expected in August 2006. An April 2006 press release from RADVISION says in part:

"RADVISION, a leading provider of multimedia conferencing and communications platforms, announces today that Compal Communications Incorporated (CCI) has licensed RADVISION's leading 3G-324M protocol toolkit, along with professional consulting services, for development of an advanced 3G cellular video Windows Mobile PDA phone.

The CCI 3G PDA phone includes an advanced video application that will allow subscribers to videoconference and participate in video communities and other revenue generating services. The phone is based on Microsoft Windows Mobile 5.0 operating system and the Intel PXA270 applications processor. RADVISION developed the video application for CCI using its 3G-324M Toolkit, in collaboration with both Microsoft and Intel."

The new Fujitsu Siemens Pocket Loox 830 incorporates video telephony capability.