



Gama Network Presents:

Gamasutra.com

Audio for Mobile Devices

By *Martin Wilde*

Gamasutra

May 29, 2003

URL: http://www.gamasutra.com/resource_guide/20030528/wilde_01.shtml

The major emphasis of audio in games over the last few years has been in the realm of console and PC platforms, and for good reason. The current generation of gaming consoles sport some pretty impressive audio support. In contrast, the audio capabilities of cell phones and other mobile platforms are limited. Fortunately the functionality of these devices is growing by leaps and bounds, and that's beginning to offer some interesting possibilities for game developers.

This article outlines the current and emerging audio technologies available for games in the mobile marketplace. First I'll present a survey of mobile audio development, authoring and player technologies from a high-level perspective. Then I'll wind up with an interview with two audio professionals who generate content for numerous mobile platforms. They discuss the trends and challenges of working in the mobile audio arena, and talk about what the future may hold.

Development Technologies

Beatnik mobileBAE (<http://www.beatnik.com/products/mobilebae.html>). Beatnik's mobileBAE (Beatnik Audio Engine) is an optimized version of its well-known engine for the minimal RAM, ROM and MIPS of mobile platforms. It is a software-based audio synthesis engine with a multi-channel mixer for resource-constrained handheld digital devices. It can manage the audio playback of Musical Instrument Digital Interface (MIDI) data and digitally recorded audio simultaneously, and supports customizable instrument banks. It is available for standard software operating systems including the Symbian OS, Windows CE, Linux and, more recently, ARM processors. The mobileBAE supports a broad range of open standards including SP-MIDI (Scalable Polyphony MIDI), DLS (Downloadable Sounds) synthesis, and XMF (eXtensible Music Format). Other supported formats include MIDI, SMAF, RMF, RTX, SMS, iMelody, WAV, AIFF, and MP3. The engine also supports the Java Mobile Media API (MMAPI) within the Java 2 Platform, Micro Edition (J2ME) platform (see below).

Is the alphabet soup thick enough for you yet? Wait, it gets better.

Helix DNA Client (<https://www.helixcommunity.org>). The Helix DNA Client is the RealOne Player media engine from RealNetworks. Real's position is that the industry needs a standardized media platform to enable application development - theirs. Using Helix, consumer electronics (CE) manufacturers, set top box providers and others can integrate the RealOne Player media engine onto a broad range of devices. They are attempting to do this through a community source licensing approach in an attempt to stimulate development efforts across the industry while maintaining standardized APIs and compatibility.

The Helix DNA Client is a media playback engine designed to support the decoding and playback of many different data types, and supports any audio or video codec through file format and decoder APIs. Real is banking that you'll use the Helix DNA Client as the core media engine inside or alongside your own products to develop applications for mobile phones, set top boxes, home gateways, personal music players, and personal computers.

The Helix DNA Client contains support in source code for the MP3 and H.263 data formats. Binary-only support is provided for: RealAudio G2, RealAudio 8, RealVideo G2, RealVideo 7, RealVideo 8 and RealVideo 9. RealNetworks is also working on support for the Ogg Vorbis format (a non-proprietary, patent-and-royalty-free, general-purpose compressed audio format from the Xiph.Org Foundation).

Java Mobile Media API (<http://java.sun.com/products/mmapi>). From the Mobile Media API (MMAPI) for J2ME document, also known as JSR-135, "many multimedia types and formats exist in today's market, and new types and formats are being introduced all the time. There are also many, diverse methods to store and deliver these various media types. J2ME devices range from cell phones with simple tone generation to PDAs and web tablets with advanced audio and video rendering capabilities." The MMAPI is an attempt to accommodate diverse configurations and multimedia processing capabilities with a high level of abstraction.

The MMAPI is targeted to fulfill the need for the control and simple manipulation of sound and multimedia for applications in mobile devices, with scalability to other J2ME devices. The MMAPI extends the functionality of the J2ME platform by providing audio, video and other time-based multimedia support to resource-constrained devices. It allows Java developers to gain access to native multimedia services available on a given device in a single, lightweight package.

The reference implementation has support for simple tone generation, tone sequencing, audio/video file playback and streaming, interactive MIDI, and audio/video capture. The MMAPI runs on the Connected Limited Device Configuration (CLDC)/Mobile Information Device Profile (MIDP) under Windows 2000.

The API is extensible, and does not define any security mechanism. Therefore, the implementations of MMAPI are subject to the security mechanisms provided by the underlying profile and configuration.

The eXtensible Music Format (<http://www.midi.org/xmf>). The eXtensible Music Format (XMF) specification was standardized by the MIDI Manufacturers Association (MMA) and its Japanese sister organization, the Association of Musical Electronics Industry (AMEI), in 2001. From the Specification for XMF Meta File Format (version 1.00b), "XMF is a low-overhead meta-file format for bundling collections of data resources (i.e. file images) in one or more formats into a single file." Put another way, XMF is an open container format in which multiple audio file types can be packaged and delivered.

A primary objective of the XMF specification is that the files be usable on all playback platforms, including small mobile devices. To achieve this, the XMF format is platform agnostic with a scalable container data structure. For instance, chunk boundaries are not constrained to word, double-word or other processor-specific boundaries. Resource sharing and dynamic Internet publishing is also facilitated by optionally referencing a resource in another XMF file or URI instead of a data block inside the local XMF file.

The XMF specification can store of custom resource types, and the application of any custom security or compression algorithms. XMF also supports the inclusion of file metadata containing information such as composer name, publisher name, copyright, and licensing details. XMF metadata can also contain unique identifiers to lock playback of a file to a specific device or user account.

To ensure a consistent playback experience across all mobile platforms, Type 0 and Type 1 XMF files specify how to bundle Standard MIDI File (SMF) and DLS file images. The DLS specification provides the performance details necessary to ensure that instrumentation and quality is consistent across playback environments. With XMF and DLS both globally adopted standards, XMF content is positioned to become the de facto audio standard for mobile devices. Enabling custom sample sets for ringtones in the future, we will no longer be tied to a GM sound set, and anything is possible.

Although the focus of the initial XMF specification is music, the format can be extended to support virtually any media type, including audio, text, and images. As Multimedia Messaging Service (MMS) services continue to make their way into the marketplace, XMF provides a simple solution for transmitting multimedia messages wirelessly in a single package.

Scalable Polyphony MIDI (http://www.midi.org/about-midi/gm/gml_spec.shtml). Scalable Polyphony MIDI (SP-MIDI) is a variant of the MIDI specification. It defines a flexible method by which an SMF can play predictably on devices with varying capabilities. These devices may vary in terms of memory, processing speed, all resulting in different polyphony capabilities. Using a special Universal Real-time System Exclusive message, composers indicate how a MIDI file should be performed by choosing which parts are to be eliminated. It is essentially a channel priority and muting scheme with note-stealing.

SP-MIDI standardization activity was driven by the close link to the telecommunication standardization work for the 3rd Generation Partnership Project (3GPP). It was conceived as a solution for 3G mobile applications and systems, as an alternative to General MIDI Lite (GM Lite, which requires a fixed 16-note polyphony). It is based on General MIDI 2 (GM2) but with a smaller sound set that is more appropriate for hand-held devices with minimal storage.

For mobile applications, SP-MIDI offers both the system operator and the mobile terminal manufacturer the flexibility to address differing customer needs. It is important that the same content will play on any phone, whether it is a lower-cost phone with only 8-note polyphony, or a higher priced model with 32-note polyphony.

That way, customers can upgrade their phones and still play all of the content that they previously obtained. They can also share content with friends and relatives who may have different configurations.

SP-MIDI also helps to mitigate situations that might occur in wireless and battery powered systems. For example, a multi-purpose SP-MIDI phone could automatically drop back from 16- to 4-notes when more power was needed for some other application. Similarly, reducing polyphony would be a reasonable means for conserving battery power.

In systems and content that don't adhere to the SP-MIDI specification, notes are arbitrarily terminated or not played at all. The decision of what notes to toss is based on the capabilities of the playback device instead of the content creator's instructions. SP-MIDI changes this and allows the artist to decide in advance which musical lines will be played in different polyphony situations. Some authoring guidelines for SP-MIDI content are included later in this article.

Embedded Audio Synthesis (<http://www.sonicnetworkinc.com/mobileaudio.asp>). Embedded Audio Synthesis (EAS) is a customizable audio software package from Sonic Network, Inc. It comprises a digital audio player, a General MIDI (GM) synthesizer, matched wavetable sound sets, and multimedia extensions for multiple ringtone formats and graphics integration. The core software is a DLS-compliant synth supporting GM, GM Lite, SP-MIDI, SMAF-MA2, CMX and XMF (more on these formats and technologies below).

The EAS architecture is scaleable, supports 8- to 64-voice polyphony, 8- to 44-kHz sampling rates, and 8- to 16-bit depth with selectable interpolation.

intent Sound System (<http://www.withintent.biz>). The intent Sound System (ISS), from the Tao Group Ltd., is an application-level framework for supporting audio functionality. It is part of Tao's larger Universal Multimedia Platform, a binary portable runtime environment comprised of 2D graphics and multimedia libraries. The ISS contains 4 primary components. They are: 1) the Audio Output Manager, or mixer, 2) the Audio Capture Manager, used to manage audio input devices, 3) the MIDI Output Manager, used to play MIDI content through either software synthesizers or a physical MIDI hardware device, and 4) the MIDI Input Manager, used by applications that are interested in receiving MIDI event information from external MIDI controller devices.

Short Message Service. Short Message Service (SMS) is a service for sending messages to mobile phones that use the Global System for Mobile (GSM) communication protocol (widely used in Europe and increasingly available in the United States). SMS is similar to paging, except that messages do not require the mobile phone to be active and within range and will be held for a number of days. SMS messages are transmitted within the same geographical area covered by a cellular telephone transmitter, or to anyone with roaming service capability. They can also be sent to digital phones from a Web site equipped with PC Link or from one digital phone to another.

Enhanced Message Service. Enhanced Messaging Service (EMS) is an adaptation of the SMS that allows users to send and receive ring tones as well as combinations of simple media to and from EMS-compliant handsets. It can use SMS centers the same way that SMS does, and works on all GSM communications networks. One specific audio enhancement carried by EMS is that melodies are transmitted according to a subset of the iMelody 1.0 format of the Infrared Data Association (IrDA). In general, if a message is sent to a phone that is not EMS-capable, the recipient will simply discard the parts of the message it doesn't understand yet still receive the text portion of the message.

EMS users can integrate text, melodies, pictures, sounds, and animations to enhance their messages that are limited by the display constraints of mobile devices. Message senders can use images, sounds and animation they download from an online library or create images and sounds directly on the phone.

EMS is an open standard developed by the 3GPP, and is being actively promoted by Alcatel, Ericsson, Motorola, and Siemens. Nokia is promoting a similar standard, called MMS.

Multimedia Messaging Service. Multimedia Messaging Service (MMS) is a way to combine text, pictures, photos, animations, speech and audio in a single message. MMS enables mobile users to send these multimedia messages from MMS-enabled handsets to other mobile and e-mail users. It also makes it possible for mobile users to receive multimedia messages from other mobile devices, e-mail and from multimedia enabled applications.

MMS builds on the successful message-push paradigm of SMS and enhances communication possibilities for mobile users by integrating the new standards from 3GPP and the Wireless Application Protocol (WAP) forum.

iMelody (<http://www.irda.org/standards/pubs/iMelody.pdf>). iMelody is an ASCII melody format. It features volume modifiers which can be used to vary the actual volume throughout a melody, and includes special codes to flash a phone's backlight, LED or make it vibrate. Some example applications include ring tones, alarm tones and power-on melodies. It has also been adopted as a ringtone format by the companies developing EMS.

Compact Media Extensions (<http://www.cdmatech.com/solutions/pdf/cmxfaq.pdf>). Compact Media Extensions (CMX) from Qualcomm is a software-based system to provide time-synchronized multi-media presentations on phones. It permits MIDI to be combined with text, graphics, animation and voice. CMX sends instructions to a built-in player in phones to play music and animations.

Content is assembled using the CMX Authoring Tool that runs on a standard PC. It's a desktop application where one builds content by dragging media elements onto timeline.

The music is powered by a wavetable synthesizer from Faith, Inc. CMX supports the inclusion of speech using a tool called PureVoice that compresses it into an 8k, 16-bit mono format. CMX supports the Portable Networks Graphics (PNG) bitmap graphics format, is compatible with Code Division Multiple access (CDMA) networks, and is not surprisingly supported by Qualcomm's Binary Runtime Environment for Wireless (BREW) API.

Authoring Technologies

SP-MIDI. Authoring SP-MIDI content requires a set of initialization messages. The primary messages are the SP-MIDI Maximum Instantaneous Polyphony (MIP) Message, and the Device Initialization message. The MIP message informs the receiving device about the polyphony requirements for each MIDI channel, and the channel priority order. The Device Initialization message sets the receiving device into the proper mode, such as GM or DLS.

Where desired, SP-MIDI content can be made to play on non-scalable devices. For example, content that conforms to the GM2 specification (32-notes) can include a MIP message to control the playback on lower-polyphony devices, and that message will be ignored and have no effect on a GM2 device. In this way it is assumed that much of the existing SMF content in use today can be re-authored for use in SP-MIDI devices, simply by giving some consideration to channel priority and masking and inserting an appropriate MIP message.

Polyphony generally refers to the number of MIDI voices simultaneously playing, which, according to the SP-MIDI Specification, refers to how many MIDI notes are on at the same time. Note that some playback devices treat voices as the number of sounds that are simultaneously present. For example, if you are using an instrument with a long release, even though the device has received a MIDI note-off message, SP-MIDI still treats the continued sound of the note as a voice in its allocation.

The MIP message specifies the total number of voices required for accurate playback of a given MIDI channel with all MIDI channels of higher priority. Each channel used in a SP-MIDI file requires a MIP value. Collectively, all the MIP values in a file make up the MIP table that represents a cumulative polyphony requirement based on each channel's priority, as defined by the content creator.

The MIP Message is a Universal Real-time System Exclusive that contains the channel priority and MIP table information. When placed at the very beginning of a MIDI sequence, the MIP message tells the device that this sequence is SP-MIDI compliant and specifies how the device should playback the sequence.

Jean Luc Borla and Hayden Porter have published an excellent tutorial on how to author an SP-MIDI file on the Sonify.org website (see the link at the end of this article). In that tutorial, they identify six steps in the process of making a MIDI file into one that is SP-MIDI compliant. They point out that an important issue is the cumulative nature of a MIP message. The MIP value for each successive channel must be at least the sum of its own polyphony and all the previous channels' polyphony.

Since writing a MIP code can be a bit technical, Beatnik has created a simple browser based tool that can streamline the process of creating MIP data. The Beatnik MIP tool (see link at the end of this article) allows one to fill-in relevant track information into a web form and it generates the appropriate MIP data for the MIDI file.

Synthetic Music Mobile Application Format (<http://smaf-yamaha.com>). The Synthetic music Mobile Application Format (SMAF) is a hardware-based, proprietary solution from Yamaha. It defines a data format for the efficient and compact representation of multimedia content for use on mobile devices. It is a data format designed for the synchronous playback of multimedia content, all contained in a single file. A chunk structure is used, allowing the score tracks (synthesizer music tracks), and other audio tracks and graphic tracks (including text display) to be described independently.

The three different flavors of SMAF (MA-1, -2 and -3) refer to different levels of polyphony. The sounds are generated using their OPL3 chipset to provide 4-operator Frequency Modulation (FM) synthesis technology. Up to 16-notes of FM and 8-notes of Yamaha ADPCM are supported. The synthesizer can also be run in 2-operator mode to yield more voices. But as any long-time gamer knows, those sounds aren't very interesting, so stick with the 4-op voices.

SMAF does not comply with MMS messaging standards. Coupled with its hardware-based solution, this limits its usefulness to terminal manufacturers in Japan and Korea.

RTTTL, RTX. The RTTTL specification (Ringing Tones Text Transfer Language) it is a text format used to transfer ringtones (melodies for mobile phones) on Nokia mobile phones only. It consists of a set of text parameters to express musical notation such as note name, duration, beats/minute, and volume.

The RTX file format is a text file containing the ringtone name, a control section and a section containing a comma separated sequence of ring tone commands. The RTX ringtone description format is designed to be backward compatible with RTTTL, but offers extensions in line with the Nokia Smart Messaging standard.

Player Technologies

Koan Interactive Audio for Pocket PC (<http://www.sseyo.com/koan/ppc>). The Koan system is a fully programmable, open, scalable interactive audio platform for Pocket PC software developers for low-bandwidth mobile audio applications. The software-only SSEYO Koan platform (now a wholly-owned subsidiary of the Tao group) can be licensed for and deployed on a wide range of operating systems, including Windows, Mac, and WindowsCE. The core Koan technology consists of text-based audio "vectors" that can include synthesizer and FX sound settings as well as musical information.

The Koan system can play vector audio, Koan files and MIDI files. Koan content can include fixed melodic sequences and patterns, and utilize and apply FX to audio samples (e.g. MP3). The Koan system can also deliver complex and evolving non-looping generative audio, which can be driven by external events, and so is appropriate for usage in games and interactive entertainment.

Beatnik Player for Pocket PC (<http://www.beatnik.com>). The Beatnik Player for PocketPC is a fully functional demonstration of a mobile application built on top of the Beatnik Audio Engine (BAE). This application can play back linear audio files, such as MP3 and WAV files, as well as MIDI and Rich Media Format (RMF) files (Beatnik's own music file format). It can also add real-time reverb effects to any music format. When playing RMF files, the player provides the ability to control and interact with RMF titles.

Windows Media Player for Pocket PC (<http://www.microsoft.com/windows/windowsmedia/download/pocket.aspx>). Windows Media Player for Pocket PC brings common digital media activities together in one application. It gives full support for downloaded digital audio and video playback, playlist management, and support for Digital Rights Management (DRM). It features automatic discovery of content on your device, integration with the desktop version of Windows Media Player and automatic synchronization. It is designed to be easy to use and deliver extremely high quality audio and video playback. Additionally, the player is fully skinnable and can be used to play files in the background while you are working on other tasks.

Version 7.1 adds support for Video 8, plus it includes automatic playlist creation and full support of secure content.

Two Perspectives On Mobile Audio

Now that we have a good grounding in the audio technologies available for mobile devices, where do we go? I asked two composers of audio content for mobile devices, David Brenner of Motorola and Jeff Essex of audiosyncrasy (a sound design firm), to get their perspectives on how this technology is best applied in the marketplace.

With all of the different formats out there (GM, GM Lite, SP-MIDI, DLS, XMF, SMAF, CMX, etc.), how does one decide what format(s) to support, and what formats to create content in?

For David Brenner of Motorola, it basically comes down to scalable solutions and what is supported within the MMS specification. For example, SP-MIDI by its very design is a scalable solution for music synthesis in a 5-24 note profile and is in the MMS and 3G specs. SMAF is a proprietary, hardware-based solution from Yamaha. It doesn't comply with MMS messaging standards, and is supported predominantly only in Korea and Japan. GM Lite is not scalable. CMX is not supported in the MMS spec.

For Jeff Essex, it really doesn't matter to him. His clients tell him what they need.



Jeff Essex

It could be based on the number of handsets out there, or what formats are supported where. But in the end, it's not his decision to make.

As far as formats and standards within the industry are concerned, there are two important developments afoot. The first is an effort to standardize multimedia content delivery in the XMF format. This would mean that both MIDI and DLS resources could be easily bundled together. The second is that the MMA is in the process of defining a Mobile DLS specification that would prescribe synthesizer performance and make other recommendations for mobile devices. Once these two efforts are finalized and make it into the MMS spec, it would be a no-brainer for artists to publish their mobile content in XMF files for the widest distribution and exposure.

Are there content creation tools for all these formats?

Jeff Essex admitted that finding tools is a challenge.

"MIDI tools abound, but there is not a lot of stuff out there for SP-MIDI yet," Essex said. "Beatnik has an editor that can perform an analysis of the number of simultaneous voices in a MIDI file, which can be useful when creating the MIP message for SP-MIDI files. They also have a MIP tool that will create the MIP message for you based on your own MIP settings.

"Similarly, there is nothing currently available to create XMF files, though there are rumblings that some sequencer manufacturers are looking to make this happen for MIDI+WAV files, at least for file-sharing and compatibility purposes.

"For SMAF, Yamaha has an FM design and audition tool for their format."

Who is supplying the content for mobile devices today?

Especially within the terminal providers, there are in-house folks (like David) that want to retain control of their own content. But there are also a number of third-party outfits (Tribal Brands, Moviso, audiosyncrasy) who are working in this area, with more developers getting in all the time. It will be interesting to see how many of these third-party providers the business will support.

The licenses for tunes are typically negotiated by the third-party provider, or already negotiated by the client, and sold to service providers or terminal makers for installation and download. And to be MMS-compliant, content should generally be delivered in SP-MIDI format, with recommended access to a GM-compliant synth.

Are there any trends or standardization efforts underway which would change or grow the pool of available content artists?

"Definitely XMF," Brenner replied. "But the greatest limitations today for mobile devices are processing bandwidth, storage limitations inside the device and the high cost of large message transmission. These directly relate to the quality of sound and cost of the device. However, when memory costs come down, processing power goes up and the costs for downloading or receiving larger messages decrease, the majority of audio on mobile devices will become increasingly sample-based, either in MP3 or some other format. Then everyone with a sampler will become a potential ringtone mix specialist, from major recording artists on down to home studio enthusiasts."

How is the customer served by the alphabet soup that is mobile audio today?

"Since content delivery is tied to the service provider, the customer can only get what's offered by their wireless company," says Essex. "The customer is ignorant of it all. They go to the mobile web site and download their new tones and songs. It's transparent to them. All they know is they're charged to install new stuff."

With such seemingly paltry resources and audio quality (small wavetables, tiny speakers and enclosures, reduced bit-depths and sample rates), why the big push for multimedia and audio on mobile devices/cell phones?

Essex and Brenner both agree that it's already getting quite a bit better, and will continue to improve by pushing the envelope. "That's always been our job," says Essex, "whether it was CD-ROM ten years ago, the web seven years ago, or wireless today. Two years ago, you couldn't get a 16-voice capable phone in the US. Now they're commonplace."

As more phones support existing authoring tools and standards, developers will have a much easier time building games and rich multimedia content. Furthermore, David adds, it sells. "It's what people want. People want it, so

carriers want it, and terminal providers have to give it. For if we don't, someone else will."

Where do you see mobile audio going in the future?

There'll be lots more streamed content coming out. We'll still need wavetable synths and MIDI for ringtones and games, until phones have processing power to perform interactive control of digital audio. But as more XMF (MIDI + DLS) content becomes supported, anyone can sell to anyone else. This will only serve to increase pool of potential content providers.

Other quality improvements for audio are also in the offing. For instance, being able to use your own headsets with any manufacturer's phone would be a huge plus. Handset makers aren't headphone specialists, so get out of that business, says David. Jeff believes stereo earbuds would also provide more immersive audio experiences.

What, if anything, should someone be doing now to start working in the mobile audio space?

"Be a proficient MIDI composer, and get familiar with SDKs for various phones," Brenner recommended.

Jeff Essex went further. "At least get yourself a 16-voice phone, sign up with a decent provider and start experimenting with the tunes and games that are already available," he said. "The skills that you may have learned from previous technology business cycles (i.e. think small, and meet the platform on its own terms) will come in handy."

It's clear that game developers creating content for mobile devices are having to cope with audio technology that is still in its nascent stages. Yet these devices contain a number of strengths and possibilities that you shouldn't ignore. The landscape of options out there can be daunting, but the future is here.

Selected Links

Sonify.org SP-MIDI tutorial

http://sonify.org/tutorials/links/pages/mobile_audio/authoring/sp-midi/

Beatnik MIP tool

<http://www.beatnik.com/developers/spmidi/miptool.html>

Pointers on how to prepare and encode audio and video files for playback on RealOne-enabled mobile devices

<http://www.realnetworks.com/resources/howto/mobile/index.html>

Tips on creating mobile media from RealNetworks

<http://www.realnetworks.com/mobile/create/index.html>

Copyright © 2003 CMP Media Inc. All rights reserved.