

METHOD FOR GENERATING AN AUDIO FILE ON A SERVER UPON A REQUEST FROM A MOBILE
PHONE

The present invention relates to the generation of sound from audio files in mobile telecommunications devices such as mobile telephones.

Nowadays, mobile phones are used not only for real-time conversation, but also for running software applications.

Mobile phone based applications are already a thriving market and will be even more so in the coming years. It is possible to perform many tasks in these applications from the popular sms (short message service) and games to recording video from a plug in camera and then sending it to another phone.

What is not possible at the moment is dedicated audio functionality i.e. the ability for dynamic creation of audio data. It will come eventually but only on the very expensive handsets and not for a while yet.

Distributed architectures can relieve handsets of dedicated functionality by having a simple client that connects across a network to a server that does all of the hard work. In this context "client" means an application using the services of a server.

The present invention is based on the proposal that a distributed architecture can provide dedicated audio functionality. This will work by the server doing the hard work of generating the required audio file and the client that resides on the phone simply downloading the resulting audio file.

Thus, the present invention provides a method of generating sound in a mobile telephone operating in a communications network including a server capable of communication simultaneously with several mobile telephones, in which an application running on the

mobile telephone causes a request for an audio file to be sent to the server, the requested audio file is generated by means within the server in response to the request, transmitted from the server to the mobile telephone and reproduced by the mobile telephone.

Some possible applications for this are as follows:

1. Text to speech
2. Music generation
3. Sound effect generation
4. Text to music
5. Audio enhancement (to be explained below)

Some embodiments of the invention will now be described by way of example only and with reference to the accompanying drawings in which:

Figure 1 is a schematic diagram of a communications network in which the present invention may be used; and

Figure 2 is a schematic diagram of a possible software architecture implementing the invention.

Figure 1 shows a mobile telephone 10 having a communication channel to a data centre 11 having server 13 via a radio access network including base transmitter stations 12 and a core network linking base stations 12 to various data centres.

In the example in Figure 2, the telephone 10 has distributed audio client software indicated at 20 and the server 13 has distributed audio server software indicated at 23.

In the specific examples which follow, the phone 10 must support the following client software:

1. Application environment; and
 2. Ability to play sound files through the application environment
- or
1. SMS capability; and
 2. Ability to playback MMS audio files

The following types of currently available handsets are examples that meet the required criteria:

1. MIDP 1.0 with audio extensions
2. DoJava 1.0 with audio extensions
3. MIDP 2.0
4. Native (proprietary) application support with audio capability

The phone will run an application 20 that acts as a client. This client will send a request to the server software 23 to perform a task, i.e. generate an audio file.

The following are example tasks:

1. Take a text-based input, convert the text to speech, and output a WAV audio file.
2. Take a text-based input that describes a music file, convert to a music file, and output a MIDI file.

3. Take an audio input and convert it to an enhanced or modified speech file, e.g. vocal with added backing music

The server 23 will read in the request, perform the required request and then send the resulting audio file back.

Preferably the server is able to support the following:

1. Receive messages by Http
2. Send message by MMS or Http
3. Create audio files

The following types of currently available server will meet the required criteria:

1. J2SE 1.3+
2. C/C++ with audio capabilities
3. PERL with audio capabilities

Use Case 1: Text to speech

1. The client 20 is written in MIDP 2.0
2. The client is used by a game which wishes to play some speech which says "Game Over"
3. The game asks the client to request the "Game Over" speech
4. The client sends a text based message to the server 23 (using TCP/IP) containing the message "Game Over"
5. The server receives the text-based message and proceeds to call a program to convert the "Game Over" text into a (8-bit, 8K, PCM, mono) WAV file.
6. The server sends the WAV file to the client

7. The client receives the WAV file and notifies the game that it has received the sound file
8. The game retrieves the sound file and plays it
9. The player hears the speech "Game Over"

The round trip time may be less than 5 seconds for small pieces of speech

Use Case 2: Text based notes to music

1. The client 20 is written in MIDP 2.0 with MIDI playback capabilities
2. The client is used by a DJ disc jockey application which wishes to generate some music
3. The DJ application asks the client to request the music and specifies a description of the music consisting of a set of notes and pauses
4. The client sends a text based message to the server 23 (using TCP/IP) containing the description of the music to generate
5. The server receives the text-based message and proceeds to call a program to convert the musical description text into a MIDI (format 0) file
6. The server sends the MIDI file to the client
7. The client receives the MIDI file and notifies the DJ application that it has received the sound file
8. The DJ application retrieves the music and plays it
9. The player hears the music

The round trip time is less than 10 seconds for small pieces of music

Use Case 3: Enhancement of audio file

Example: Sing like Barry White

[MMS = Multimedia Messaging]

1. The handset supports MMS
2. A user records himself singing
3. The user attaches the audio clip and the text "Barry White" to a MMS
4. The user sends the MMS to the text-to-speech server
5. The server receives the MMS and extracts the audio sample and the textual parameter
6. The server adds audio effects to the audio clip based on the text parameter
7. The server in this case adds audio effects to the audio clip to make the users voice sound like Barry White
8. The server attaches the resulting altered audio clip to a new MMS
9. The server sends the MMS back to the user
10. The user receives the MMS and play it to hear himself singing like Barry White
11. The user receives the MMS asynchronously so the round trip time is not important. Should be less than 30 seconds.

Glossary

SMS	short message service
MMS	multimedia messaging
MIDP	mobile information device profile
Java	a platform independent programming language
MIDI	musical instrument digital interface
Http	hypertext transfer protocol
WAV	file type for audio files
TCP	transmission control protocol
IP	internet protocol
RAN	radio access network

Claims

1. A method of generating sound in a mobile telephone operating in a communications network including a server capable of communication simultaneously with several mobile telephones, in which an application running on the mobile telephone causes a request for an audio file to be sent to the server, the requested audio file is generated by means within the server in response to the request, transmitted from the server to the mobile telephone and reproduced by the mobile telephone.
2. A method as claimed in claim 1 in which request data is generated from text input by the telephone user via the keypad.
3. A method as claimed in claim 2 in which the request data is used by means at the server to generate a speech file.
4. A method as claimed in claim 2 in which the requested data is used by means at the server to generate a music file.
5. A method as claimed in any preceding claim in which the request includes an audio file.
6. A method as claimed in claim 5 in which the audio file in the request is incorporated in the audio file generated by the server.