(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
7 April 2016 (07.04.2016)

(10) International Publication Number
WO 2016/053432 A1

(51) International Patent Classification:
G06F 17/17 (2006.01) *H04R 29/00* (2006.01)
G06F 17/10 (2006.01)

(21) International Application Number:
PCT/US2015/038635

(22) International Filing Date:
30 June 2015 (30.06.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
14/506,187 3 October 2014 (03.10.2014) US

(71) Applicant: **DTS, INC.** [US/US]; 5220 Las Virgenes Road, Calabasa, CA 91302 (US).

(72) Inventors: **STEIN, Edward**; 1710 44th Avenue, Apt. A, Capitola, CA 95010 (US). **WALSH, Martin**; 20 Milano Court, Scotts Valley, CA 95066 (US). **KELLY, Michael**; 18 Hopedale Road, London (GB).

(74) Agent: **FISCHER, Craig**; DTS, Inc., 5220 Las Virgenes Road, Calabasas, CA 91302 (US).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

(54) **Title**: DIGITAL AUDIO FILTERS FOR VARIABLE SAMPLE RATES

(57) **Abstract**: Various exemplary embodiments relate to a method and apparatus for processing audio signals to influence the reproduction of the audio signals. The apparatus may include a speaker, a headphone (over-the-ear, on-ear, or in-ear), a microphone, a computer, a mobile device, a home theater receiver, a television, a Blu-ray (BD) player, a compact disc (CD) player, a digital media player, or the like. The apparatus may be configured to receive a virtualization profile including a digital audio filter with a design sample rate, resample the virtualization profile to a different sample rate, filter the audio signal with the resampled virtualization profile, and reproduce the filtered audio signal as sound.

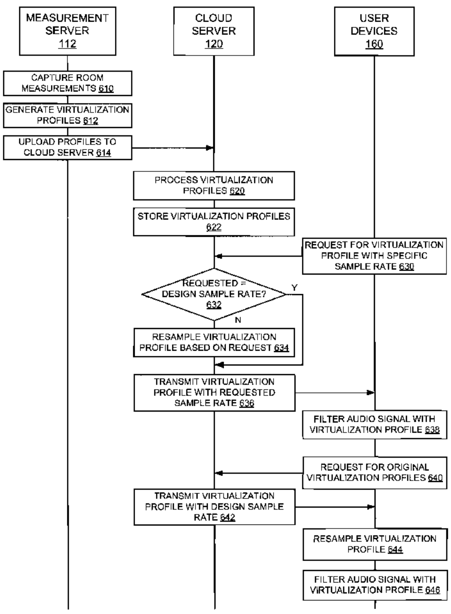


FIG. 6

WO 2016/053432 A1

DIGITAL AUDIO FILTERS FOR VARIABLE SAMPLE RATES

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Patent Application Serial Number 14/506,187, filed October 3, 2014, entitled "DIGITAL AUDIO FILTERS FOR VARIABLE SAMPLE RATES", the entire contents of which is hereby incorporated herein by reference.

BACKGROUND

[0002] In traditional audio reproduction, digital audio filters are designed for particular sample rates. If audio is reproduced at a different sample rate than the designed sample rate, digital audio filters must be redesigned or the applied filter effects need to be scaled in frequency. It would therefore be desirable to have a method and apparatus to deliver a flexible filter design that adapts to variable sample rates.

SUMMARY

[0003] A brief summary of various exemplary embodiments is presented. Some simplifications and omissions may be made in the following summary, which is intended to highlight and introduce some aspects of the various exemplary embodiments, but not to limit the scope of the invention. Detailed descriptions of a preferred exemplary embodiment adequate to allow those of ordinary skill in the art to make and use the inventive concepts will follow in later sections.

[0004] Various exemplary embodiments relate to a method and apparatus for processing audio signals to influence the reproduction of the audio signals. The apparatus may include a speaker, a headphone (over-the-ear, on-ear, or in-ear), a microphone, a computer, a mobile device, a home theater receiver, a television, a Blu-ray (BD) player, a compact disc (CD) player, a digital media player, or the like. The apparatus may be configured to receive a virtualization profile including a digital audio filter with a design sample rate, resample the virtualization profile to a different sample rate, filter the audio signal with the resampled virtualization profile, and reproduce the filtered audio signal as sound.

[0005] Various exemplary embodiments further relate to a method for processing an audio signal to influence the reproduction of the audio signal, the method comprising: sending a request to a server computer for a virtualization profile, wherein the request specifies a requested sample rate for the virtualization profile, and wherein the virtualization profile defines a digital audio filter; receiving from the server computer the virtualization profile with the requested sample rate; and filtering the audio signal based on at least the virtualization profile by performing a convolution of the audio signal with the virtualization profile with the requested sample rate.

[0006] In some embodiments, the virtualization profile represents an acoustic model of a production environment. In some embodiments, the method further comprises causing the audio signal to be reproduced as sound through an audio transducer.

[0007] Various exemplary embodiments further relate to a method for processing an audio signal to influence the reproduction of the audio signal, the method comprising: requesting a virtualization profile from a server computer, wherein the virtualization profile defines a digital audio filter; receiving from the server computer the requested virtualization profile with a design sample rate; resampling the virtualization profile at a required sample rate for the audio signal, responsive to a difference between the required sample rate and the design sample rate; and filtering the audio signal based on at least the virtualization profile with the required sample rate.

[0008] In some embodiments, resampling the virtualization profile comprises: interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and resampling the CBIR at the required sample rate. In some embodiments, filtering the audio signal comprises performing a convolution of the audio signal with the virtualization profile with the required sample rate. In some embodiments, the method further comprises causing the audio signal to be reproduced as sound through an audio transducer simulating a production environment.

[0009] Various exemplary embodiments further relate to a method for influencing reproductions of audio signals with virtualization profiles, the method comprising: storing a virtualization profile with a design sample rate, wherein the virtualization profile defines a digital audio filter; receiving a request for the virtualization profile from a client device, wherein the request specifies a requested sample rate for the virtualization profile; resampling, by a computer processor, the stored virtualization profile at the requested sample rate, responsive to a difference between the requested sample rate and the design sample rate; and transmitting the virtualization profile with the requested sample rate to the client device.

[0010] In some embodiments, the digital audio filter represents an acoustic model of a production environment comprising at least one of a finite impulse response (FIR) filter, an infinite impulse response (IIR) filter, and a feedback delay network (FDN) filter. In some embodiments, the virtualization profile causes the audio signal to be reproduced through an audio transducer simulating the production environment. In some embodiments, the virtualization profile is stored as a series of filter coefficients in fixed point or float point values. In some embodiments, resampling the virtualization profile comprises: interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and resampling the CBIR at the requested sample rate. In some embodiments, resampling the CBIR at a lower sample rate than the design sample rate results in fewer filter coefficients, and resampling the CBIR at a higher sample rate than the design sample rate results in more filter coefficients. In some embodiments, the method further comprises scaling the virtualization profile to a different sample rate to achieve a subjective audio effect.

[0011] Various exemplary embodiments further relate to a non-transitory computer-readable storage medium storing computer-executable instructions that when executed cause one or more processors to perform operations comprising: storing a virtualization profile with a design sample rate, wherein the virtualization profile defines a digital audio filter; receiving a request for the virtualization profile from a client device, wherein the request specifies a requested sample rate for the virtualization profile; resampling the stored virtualization profile at the requested sample rate, responsive to a difference between the requested sample rate and the design sample rate; and transmitting the virtualization profile with the requested sample rate to the client device.

[0012] In some embodiments, the digital audio filter represents an acoustic model of a production environment comprising at least one of a finite impulse response (FIR) filter, an infinite impulse response (IIR) filter, and a feedback delay network (FDN) filter. In some embodiments, the virtualization profile causes the audio signal to be reproduced through an audio transducer simulating the production environment. In some embodiments, resampling the virtualization profile comprises: interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and resampling the CBIR at the requested sample rate.

[0013] Various exemplary embodiments further relate to an audio device for processing an audio signal, the audio device comprising: a communication interface configured for sending a request to a server computer for a virtualization profile, wherein the request specifies a requested sample rate for the virtualization profile, and wherein the virtualization profile defines a digital audio filter simulating a virtualized environment; and receiving from the server computer the requested virtualization profile with the requested sample rate; a storage device for storing the received virtualization profile; and a processor in communication with the storage device and the communication interface, the processor programmed for filtering the audio signal based on at least the virtualization profile by performing a convolution of the audio signal with the virtualization profile with the requested sample rate.

[0014] Various exemplary embodiments further relate to an audio device for processing an audio signal, the audio device comprising: a communication interface configured for requesting a virtualization profile from a server computer, wherein the virtualization profile defines a digital audio filter simulating a virtualized environment; and receiving from the server computer the requested virtualization profile with a design sample rate; a storage device for storing the received virtualization profile; and a processor in communication with the storage device and the communication interface, the processor programmed for resampling the virtualization profile at a required sample rate for the audio signal, responsive to a difference between the required sample rate and the design sample rate; and filtering the audio signal based on at least the virtualization profile with the required sample rate.

[0015] In some embodiments, resampling the virtualization profile comprises: interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and resampling the CBIR at the required sample rate. In some embodiments, filtering the audio signal comprises performing a convolution of the audio signal with the virtualization profile at the required sample rate.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] These and other features and advantages of the various embodiments disclosed herein will be better understood with respect to the following description and drawings, in which like numbers refer to like parts throughout, and in which:

[0017] FIG. 1 is a high-level block diagram illustrating an example environment 100 for cloud-based digital audio virtualization service, according to one embodiment.

[0018] FIG. 2 is a block diagram illustrating components of an example computer system for cloud-based digital audio virtualization service, according to one embodiment;

[0019] FIG. 3 is a block diagram illustrating functional modules within a cloud server for the cloud-based digital audio virtualization service, according to one embodiment.

[0020] FIG. 4A is a block diagram illustrating the bandlimiting effect of the CBIR resampling at a lower rate than the design sample rate, according to one embodiment.

[0021] FIG. 4B is a block diagram illustrating the bandlimiting effect of the CBIR resampling at a higher rate than the design sample rate, according to one embodiment.

[0022] FIG. 5 is a block diagram illustrating functional modules within a user device for the cloud-based digital audio virtualization service, according to one embodiment.

[0023] FIG. 6 is a detailed interaction diagram illustrating an example process for providing cloud-based digital audio virtualization, according to one embodiment.

DETAILED DESCRIPTION

[0024] The detailed description set forth below in connection with the appended drawings is intended as a description of the presently preferred embodiment of the invention, and is not intended to represent the only form in which the present invention may be constructed or utilized. The description sets forth the functions and the sequence of steps for developing and operating the invention in connection with the illustrated embodiment. It is to be understood, however, that the same or equivalent functions and sequences may be accomplished by different embodiments that are also intended to be encompassed within the spirit and scope of the invention. It is further understood that the use of relational terms such as first and second, and the like are used solely to distinguish one entity from another entity without necessarily requiring or implying any actual such relationship or order between such entities.

[0025] A sound wave is a type of pressure wave caused by the vibration of an object that propagates through a compressible medium such as air. A sound wave periodically displaces matter in the medium (e.g. air) causing the matter to oscillate. The frequency of the sound wave describes the number of complete cycles within a period of time and is expressed in Hertz (Hz). Sound waves in the 12 Hz to 20,000 Hz frequency range are audible to humans.

[0026] The present application concerns a method and apparatus for processing audio signals, which is to say signals representing physical sound. These signals may be represented by digital electronic signals. In the discussion which follows, analog waveforms may be shown or discussed to illustrate the concepts; however, it should be understood that typical embodiments of the invention may operate in the context of a time series of digital bytes or words, said bytes or words forming a discrete approximation of an analog signal or (ultimately) a physical sound. The discrete, digital signal may correspond to a digital representation of a periodically sampled audio waveform. As is known in the art, for uniform sampling, the waveform may be sampled at a rate at least sufficient to satisfy the Nyquist sampling theorem for the frequencies of interest. For example, in a typical embodiment a uniform sampling rate of approximately 44.1kHz may be used. Higher sampling rates such as 96kHz or 192kHz may alternatively be used. The quantization scheme and bit resolution may be chosen to satisfy the requirements of a particular application, according to principles well known in the art. The techniques and apparatus of the invention typically would be applied interdependently in a number of channels. For example, it may be used in the context of a "surround" audio system (having more than two channels).

[0027] As used herein, a "digital audio signal" or "audio signal" does not describe a mere mathematical abstraction, but instead denotes information embodied in or carried by a physical medium capable of detection by a machine or apparatus. This term includes recorded or transmitted signals, and should be understood to include conveyance by any form of encoding, including pulse code modulation (PCM), but not limited to PCM. Outputs or inputs, or indeed intermediate audio signals may be encoded or compressed by any of various known methods, including MPEG, ATRAC, AC3, or the proprietary methods of DTS, Inc. as described in U.S. patents 5,974,380; 5,978,762; and 6,487,535. Some modification of the calculations may be required to accommodate that particular compression or encoding method, as will be apparent to those with skill in the art.

[0028] The present invention may be implemented in a consumer electronics device, such as a Digital Video Disc (DVD) or Blu-ray Disc (BD) player, television (TV) tuner, Compact Disc (CD) player, handheld player, Internet audio/video device, a gaming console, a mobile phone, or the like. A consumer electronic device includes a Central Processing Unit (CPU) or Digital Signal Processor (DSP), which may represent one or more conventional types of such processors, such as an IBM PowerPC, Intel Pentium (x86) processors, and so forth. A Random Access Memory (RAM) temporarily stores results of the data processing operations performed by the CPU or DSP, and is interconnected thereto typically via a dedicated memory channel. The consumer electronic device may also include permanent storage devices such as a hard drive, which are also in communication with the CPU or DSP over an I/O bus. Other types of storage devices, such as tape drives and optical disk drives, may also be connected. A graphics card is also connected to the CPU via a video bus, and transmits signals representative of display data to the display monitor. External peripheral data input devices, such as a keyboard or a mouse, may be connected to the audio reproduction system over a USB port. A USB controller translates data and instructions to and from the CPU for external peripherals connected to the USB port. Additional devices such as printers, microphones, speakers, and the like may be connected to the consumer electronic device.

[0029] The consumer electronic device may utilize an operating system having a graphical user interface (GUI), such as WINDOWS from Microsoft Corporation of Redmond, Washington, MAC OS from Apple, Inc. of Cupertino, CA, various versions of mobile GUIs designed for mobile operating systems such as Android, and so forth. The consumer electronic device may execute one or more computer programs. Generally, the operating system and computer programs are tangibly embodied in a computer-readable medium, e.g. one or more of the fixed and/or removable data storage devices including the hard drive. Both the operating system and the computer programs may be loaded from the aforementioned data storage devices into the RAM for execution by the CPU. The computer programs may comprise instructions which, when read and executed by the CPU, cause the same to perform the steps to execute the steps or features of the present invention.

[0030] The present invention may have many different configurations and architectures. Any such configuration or architecture may be readily substituted without departing from the scope of the present invention. A person having ordinary skill in the art will recognize the above described sequences are the most commonly utilized in computer-readable mediums, but there are other existing sequences that may be substituted without departing from the scope of the present invention.

[0031] Elements of one embodiment of the present invention may be implemented by hardware, firmware, software or any combination thereof. When implemented as hardware, the audio codec may be employed on one audio signal processor or distributed amongst various processing components. When implemented in software, the elements of an embodiment of the present invention may be the code segments to perform various tasks. The software may include the actual code to carry out the operations described in one embodiment of the invention, or code that may emulate or simulate the operations. The program or code segments can be stored in a processor or machine accessible medium or transmitted by a computer data signal embodied in a carrier wave, or a signal modulated by a carrier, over a transmission medium. The "processor readable or accessible medium" or "machine readable or accessible medium" may include any medium configured to store, transmit, or transfer information.

[0032] Examples of the processor readable medium may include an electronic circuit, a semiconductor memory device, a read only memory (ROM), a flash memory, an erasable ROM (EROM), a floppy diskette, a compact disk (CD) ROM, an optical disk, a hard disk, a fiber optic medium, a radio frequency (RF) link, etc. The computer data signal includes any signal that may propagate over a transmission medium such as electronic network channels, optical fibers, air, electromagnetic, RF links, etc. The code segments may be downloaded via computer networks such as the Internet, Intranet, etc. The machine accessible medium may be embodied in an article of manufacture. The machine accessible medium may include data that, when accessed by a machine, may cause the machine to perform the operation described in the following. The term "data" here refers to any type of information that may be encoded for machine-readable purposes. Therefore, it may include program, code, data, file, etc.

[0033] All or part of an embodiment of the invention may be implemented by software. The software may have several modules coupled to one another. A software module may be coupled to another module to receive variables, parameters, arguments, pointers, etc. and/or to generate or pass results, updated variables, pointers, etc. A software module may also be a software driver or interface to interact with the operating system running on the platform. A software module may also be a hardware driver to configure, set up, initialize, send and receive data to and from a hardware device.

[0034] One embodiment of the invention may be described as a process which is usually depicted as a flowchart, a flow diagram, a structure diagram, or a block diagram. Although a block diagram may describe the operations as a sequential process, many of the operations may be performed in parallel or concurrently. In addition, the order of the operations may be re-arranged. A process may be terminated when its operations are completed. A process may correspond to a method, a program, a procedure, etc.

Overview

[0035] Embodiments of the present invention provide a system and a method for cloud-based digital audio virtualization. The method and system is organized around a cloud computing platform configured to aggregate, manage, and distribute virtualization profiles of the audio content. The virtualization profiles are generally derived from acoustic measurements of the production environment and uploaded to the cloud server. When a listener plays back certain audio content with a user device, the user device may request a corresponding virtualization profile from the cloud server and apply the virtualization profile to the audio content to reproduce the audio content with desired production properties.

[0036] FIG. 1 is a high-level block diagram illustrating an example environment 100 for cloud-based digital audio virtualization service, according to one embodiment. The service environment 100 comprises a measurement room 110, a measurement server 112, a cloud server 120, a network 140, and user devices 160. Communication between the measurement server 112, user devices 160, and cloud server 120 is enabled by network 140. The network 140 is typically a content delivery network (CDN) built on the Internet, but may include any network, including but not limited to a LAN, a MAN, a WAN, a mobile wired or wireless network, a private network, or a virtual private network.

[0037] The acoustic measurements for the virtualization profiles may be taken in rooms containing high fidelity audio equipment, for example, a mixing studio or a listening room. The room may include multiple loudspeakers, and the loudspeakers may be arranged in traditional speaker layouts, such as stereo, 5.1, 7.1, 11.1, or 22.2 layouts. Other non-standard or custom speaker layouts or arrays may also be used. Measurement room 110 shown in FIG. 1 contains a traditional 5.1 surround arrangement, including a left front loudspeaker, a right front loudspeaker, a center front loudspeaker, a left surround loudspeaker, a right surround loudspeaker, and a subwoofer. While a mixing studio having surround loudspeakers is provided as an example, the measurements may be taken in any desired location containing one or more loudspeakers.

[0038] The room acoustics measurement is conducted under the control of measurement server 112. For example, measurement server 112 may send one or more test signals to the one or more loudspeakers inside measurement room 110. The test signals may include a frequency sweep or chirp signal. Alternatively or in addition, a test signal may be a noise sequence such as a Golay code or a maximum length sequence. The acoustic measurements may be obtained by placing a measurement apparatus in an optimal listening position, such as a producer's chair. The measurement apparatus may be a free-standing microphone, binaural microphones placed within a dummy head, or binaural microphones placed within a test subject's ears. As each loudspeaker plays the test signal, the measurement apparatus may record the audio signal received at the listening position and transfer the measurement data to server 112. From the recorded audio signals, measurement server 112 can generate a room measurement profile for each speaker location and each microphone of the measurement apparatus. Additional room measurements may be taken at other locations or orientations in the room, for example, at an “out of sweetspot” position. The “out of sweetspot” measurements may aid in determining the acoustics of measurement room 110 for listeners not in the optimal listening position or improving the acoustic models of the room space including the optimal listening position.

[0039] In one embodiment, the virtualization profiles generated by measurement server 112 may be separated into digital audio filters, such as head-related transfer function (HRTF) and binaural room impulse response (BRIR), and/or room equalization (EQ), or other independently modeled characteristics such as early room response or late reverberation. The HRTF and BRIR filters characterize how the measurement apparatus received the sound from each loudspeaker independent of the acoustic effects of the room. The early room response characterizes the early reflections after the sound from each loudspeaker has reflected off the surfaces of the room, while the late reverberation characterizes the sound in the room after the early reflections. The HRTF and BRIR filters may be digitized and stored as audio filter coefficients, and the room EQ can be represented by acoustic models that recreate the acoustics of the room. Similarly, the early and late reverberation can be digitized as audio filter coefficients or acoustic models for simulation. Both the HRTF or BRIR filters, room EQs and other acoustic models may be transmitted and/or stored as part of the virtualization profile.

[0040] Measurement server 112 may process the virtualization profiles before uploading the virtualization profiles to cloud server 120. The processing of the virtualization profile by the measurement server 112 includes, but not limited to, validating, aggregating, summarizing, sorting, encoding, encrypting, and compressing, among other processing jobs.

The virtualization profiles processed by measurement server 112 are then uploaded to cloud server 120 for distribution. Cloud server 120 maintains the virtualization profiles, which include the full room measurement data and/or the HRTF filter coefficients, early room response parameters, and late reverberation parameters for one or more measurement rooms and one or more listening positions within each measurement room. The virtualization profiles may further include other information, such as headphone frequency response information, headphone identification information, measured loudspeaker layout information, playback mode information, measurement location information, measurement equipment information, and/or licensing/ownership information.

[0041] Cloud server 120 stores, manages, and distributes virtualization profiles for the associated audio content. The virtualization profiles can be stored and distributed as metadata that is included in channel based or object based audio bitstream. In this case, the virtualization profile may be embedded or multiplexed in a file header of the audio content, or in any other portion of an audio file or frame. The virtualization data may also be repeated in multiple frames of the audio bitstream. Alternatively or in addition, the virtualization profiles can be requested and downloaded separately from associated audio content as independent data packages. The virtualization profiles may be transferred to the user devices 160 together with the requested audio content or may be transferred separately from the audio content.

[0042] When a virtualization profile is requested by user devices 160 for certain audio content, cloud server 120 may need to process the virtualization profile before transmitting the virtualization profile to user devices 160. The processing of the virtualization profiles at cloud server 120 includes, but not limited to, searching, ranking, decoding, decrypting, resampling, and decompressing, among other processing jobs. For example, after receiving a request for virtualization profiles from user devices 160, cloud server 120 may search for virtualization profiles that match identifiers, associated audio content, or any other identification information in the request. In case more than one virtualization profiles are found, cloud server 120 may rank the search result and/or send the list of the profiles to user devices 160 for selection. If the requested virtualization profiles are encoded, encrypted, or compressed, cloud server 120 can decode, decrypt, or decompress the virtualization profile at the request of user devices 160.

[0043] In some embodiments, a virtualization profile stored at cloud server 120 is measured by the measurement server 112 at a design sample rate, for example, 48 kHz. If a request from user devices 160 for the virtualization profile with a different requested sample rate than the designed sample rate, cloud server 120 may need to resample the virtualization profile in response to the request. Details of the resampling process are further described below in reference to FIG. 4. In alternate embodiments, the resampling of the virtualization profile can also be performed by the user devices 160 if so desired and indicated by the user devices 160 upon request.

[0044] The user devices 160 are any playback or accessory devices that can compute, communicate, and render an audio signal with corresponding virtualization profiles. The user devices 160 include, for example, a headphone 162, a smartphone 164, and a laptop computer 164. Although only three user devices 162, 164 and 166 are shown in FIG. 1, any number of user devices 160, such as personal computers (PCs), tablet PC, mobile devices, set-top boxes (STBs), web appliances, network routers, switches or bridges, or audio/video systems, may communicate with cloud server 120 to acquire virtualization profiles for virtualized playback. In one embodiment, a user may be associated with an account on cloud server 120, and virtualization profiles downloaded/purchased by the user are available through all the user devices associated with the user account.

[0045] In one embodiment, when an audio content starts to play on a user device, the audio content may include a flag in its bitstream indicating the user device about available virtualization profiles at cloud server 120 for download/purchase. Once a virtualization profile is downloaded, the user device may process the virtualization profile, for example, resample the virtualization profile to match the digital audio sample rate at the user device. The processed virtualization profile is then applied to filter the audio content for virtualized listening experience. For example, an audio content may be processed in a mixing studio (e.g., measurement room 110), allowing the audio producer to measure the spatialized headphone mix that end-user hears. Later, headphone 162 may download from cloud server 120 and store a virtualization profile generated by measurement server 112 from the measurement for the audio content. If headphone 162 applies the virtualization profile to the audio content, the acoustics and loudspeaker locations of the measured room will be recreated, and the audio content will sound similar to audio played back over the loudspeakers in the measured mixing studio.

Computer Architecture

[0046] FIG. 2 is a block diagram illustrating components of an example computer able to read instructions from a computer-readable medium and execute them in a processor (or controller) to implement the disclosed system for cloud-based digital audio virtualization service. Specifically, FIG. 2 shows a diagrammatic representation of a machine in the example form of a computer 200 within which instructions 235 (e.g., software) for causing the computer to perform any one or more of the methods discussed herein may be executed. In various embodiments, the computer operates as a standalone device or connected (e.g., networked) to other computers. In a networked deployment, the computer may operate in the capacity of a server machine or a client machine in a server-client network environment, or as a peer machine in a peer-to-peer (or distributed) network environment.

[0047] Computer 200 is such an example for use as measurement server 112, cloud server 120, and user devices 160 in cloud-based digital audio virtualization environment 100 shown in FIG. 1. Illustrated are at least one processor 210 coupled to a chipset 212. The chipset 212 includes a memory controller hub 214 and an input/output (I/O) controller hub 216. A memory 220 and a graphics adapter 240 are coupled to memory controller hub 214. A storage unit 230, a network adapter 260, and input devices 250, are coupled to the I/O controller hub 216. Computer 200 is adapted to execute computer program instructions 235 for providing functionality described herein. In the example shown in FIG. 2, executable computer program instructions 235 are stored on the storage unit 230, loaded into the memory 220, and executed by the processor 210. Other embodiments of computer 200 may have different architectures. For example, memory 220 may be directly coupled to processor 210 in some embodiments.

[0048] Processor 210 includes one or more central processing units (CPUs), graphics processing units (GPUs), digital signal processors (DSPs), application specific integrated circuits (ASICs), radio-frequency integrated circuits (RFICs), or any combination of these. Storage unit 230 comprises a non-transitory computer-readable storage medium 232, including a solid-state memory device, a hard drive, an optical disk, or a magnetic tape. The instructions 235 may also reside, completely or at least partially, within memory 220 or within processor 210's cache memory during execution thereof by computer 200, memory 220 and processor 210 also constituting computer-readable storage media. Instructions 235 may be transmitted or received over network 140 via network interface 260.

[0049] Input devices 250 include a keyboard, mouse, track ball, or other type of alphanumeric and pointing devices that can be used to input data into computer 200. The graphics adapter 212 displays images and other information on one or more display devices, such as monitors and projectors (not shown). The network adapter 260 couples the computer 200 to a network, for example, network 140. Some embodiments of the computer 200 have different and/or other components than those shown in FIG. 2. The types of computer 200 can vary depending upon the embodiment and the desired processing power. Furthermore, while only a single computer is illustrated, the term "computer" shall also be taken to include any collection of computers that individually or jointly execute instructions 235 to perform any one or more of the methods discussed herein.

Virtualization Profile Resampling

[0050] Cloud server 120 stores virtualization profiles uploaded by measurement server 112, and distribute the virtualization profiles to user devices 160 over network 140. FIG. 3 is a block diagram illustrating functional modules within a cloud server 120 for the cloud-based digital audio virtualization service. In one embodiment, cloud server 120 comprises a profile manager 310, a profile database 320, a profile-processing module 330, and a network interface 340. As used herein, the term “module” refers to a hardware and/or software unit used to provide one or more specified functionalities. Thus, a module can be implemented in hardware, software or firmware, or a combination of thereof. Other embodiments of could server 120 may include different and/or fewer or more modules.

[0051] The profile manager 310 receives measurement data 305 uploaded by the measurement server 112. The measurement data 305 may include raw room measurements and/or virtualization profiles processed by the measurement server 112. The profile manager 310 may also validate, encode, encrypt and compress the measurement data 305 before storing the processed measurement data 305 in the profile database 320. For example, a room response measurement (e.g., room 110) may be sampled at 192kHz and encoded using

64 bit analog-to-digital converter (A/D) by the measurement server 112 and/or the profile manager 310. The resulting filter coefficients are then stored at the profile database 320 as a virtualization profile associated with the measurement room 110.

[0052] When stored at profile database 320, a virtualization profile may be indexed and retrieved by a unique identifier, such as an MD5 checksum or any hash values generated by other hash functions. The unique identifiers of the virtualization profiles can also be derived from other identifying information, such as measurement room information, measured loudspeaker layout information, measurement location information, measurement equipment information, associated audio content identifiers, and/or licensing and ownership information. The virtualization profile identifiers can be generated by the profile manager 310 or received from the measurement server 112. In addition to the virtualization profiles, the profile database 320 may also store other audio production profiles, such as playback device profiles and listener hearing profiles.

[0053] Requests for virtualization profiles from user devices 160 are handled by the network interface 340. A profile request 335 may include identification information of the requested virtualization profile. For example, the request 335 can specify a unique profile identifier, or other identification information such as associated audio content, measurement rooms, and/or production or license owners. The profile request 335 may further specify parameters, such as sample rate, A/D length, and bitrate, of the virtualization profile requested. The request 335 may be generated by the user devices 160 automatically based on preconfigured user device profiles and/or user preferences. Alternatively or in addition, the network interface 340 can provide a graphic user interface (GUI), such as a webpage, to the user devices 160 and prompt users to fill in identification information or other parameters of the requested virtualization profiles in advance or on the fly.

[0054] In some embodiments, the network interface 340 passes requests for virtualization profiles to the profile manager 310, which searches the requested profiles from the profile database 320. Search results are then forwarded to the profile-processing module 330. A search result may include more than one virtualization profiles, for example, a list of profiles of multiple rooms' measurements. The profile-processing module 330 may choose one or more profiles from the search results and pass the resulting virtualization profiles to the network interface 340, which transmits the virtualization profiles as a response 345 to the profile request 335 back to the requesting user device 160. In some embodiments, the profile database may also log types of virtualization profiles, number of requests, among other preference data. This may allow the cloud server to provide customized recommendations to users based on usage history.

[0055] In one embodiment, the profile-processing module 330 helps select which room's acoustics should be returned to the requesting user. For instance, the user may prefer audio content to be processed with a virtualization profile that is most similar to the acoustics of his or her current room. In this case, the profile-processing module 330 may need to communicate with the client devices 160 for acoustic measurement of the user's room with one or more tests. For example, the user may clap his or her hands in the current room, and the hand clap is recorded and processed to determine the acoustic parameters of the room. Alternatively or in addition, other environmental sounds, such as speech, may be analyzed. The tests can be processed either by the cloud server 120 or at the client devices 160. In alternate embodiments, the profile-processing module 330 may simply respond to the user request with one or more virtualization profiles and let the requesting user to select.

[0056] In many occasions, users may request virtualization profiles or filters with a different sample rate than the design sample rate captured by the measurement server 112 and/or stored at the profile database 320. One solution is to design the filters, such as an infinite impulse response (IIR) Butterworth filter, with simple operations to be automated on the fly. However, such "simple" designs often require operations (e.g., sin/cos or log) not consistent or available with high enough precision across all platforms. This lack of precision is further compounded when coupling with fixed-point systems. For the cloud-based virtualization filters (e.g., HRTFs, BRIRs, and Room EQs) derived from measured room responses, there may not be sufficient measurement data to retrieve the needed sample rate in a run-time environment.

[0057] Another option is to design filters at all possible sample rates offline and store the filters in a database by predicting and measuring room response for every sample rate that might be needed. This method may consume a significant amount of memory and storage because multiple filters are needed and each filter with a number of sample rates, making it unacceptable especially for embedded systems. A third option is to distribute both audio content and filters at the same sample rate. However, it is impractical to require numerous audio applications across various software platforms to process digital audio and filters all at a specific sample rate. Fixing audio sample rate may also cause portability problem and licensing issues. A requirement for end users to clock their global audio path to a fixed sample rate may be prohibitive in terms of computing resources, such as CPU power, memory consumption, and battery life, among other bill of materials cost.

[0058] The preferred solution is to design filters containing spectral resolution suitable for any sample rate and automatically adapted to any playback rate on the fly. It is the well-known that a sampled signal is bandlimited to half of the sampling rate (i.e., the Nyquist frequency). Shannon's sampling theorem also suggests that the original signal can be exactly and uniquely reconstructed by interpolating between the sampled values if the sampling rate is higher than the Nyquist frequency. Therefore, the method of bandlimited interpolation, which operates on the foundation Nyquist-Shannon sampling theorem, provides a means of reproducing a continuous-time, yet bandlimited impulse responses from room measurements, rather than "filter coefficients" only at a design sample rate.

[0059] In one embodiment, the profile-processing module 330 resamples the virtualization profiles to match the requested sample rate. The resampling of the virtualization profile involves applying interpolation on the virtualization profile, such as HRTF and BRIR filters, to obtain a continuous bandlimit impulse response (CBIR). The interpolated CBIR is then resampled at the requested sample rate before transmitting to the user devices 160. The interpolated CBIRs and resampled virtualization profiles can be stored and/or cached by cloud server 120 for further use if storage space allows. As a result, the method allows filters to be designed once, and later adjusted to any requested sample rates without dependency on any special functions that might deviate due to different implementations. Such a design not only maintains consistent audio fidelity across various platforms, but also simultaneously minimizes memory footprint and allows scalable processing at user devices.

[0060] The bandlimited interpolation fits perfectly to the audio filter design choice because the audio frequencies of interest lie in the audible range of 20Hz-20kHz. To measure or model an audio filter for "continuous-time", one only needs to sample the impulse response at 40kHz or higher to minimize memory and/or storage space dedicated towards filter taps. At run time, these filter taps can be interpolated and resampled at any rate to cover the spectrum of interest. For instance, if the interpolated CBIR is resampled at a rate higher than the design sample rate, an input audio signal passing the CBIR is automatically "bandlimited" at the original Nyquist frequency of the filter (i.e. 20kHz). Whereas in case of a lower resample rate, bandlimited interpolation becomes effectively a low-pass filter for the CBIR with a cutoff frequency at Nyquist frequency of the lower resample rate. In the latter case, loss of the filter specification for higher frequencies is acceptable because those frequencies are absent from the input audio signal being processed in the first place.

[0061] FIG. 4A is a diagram illustrating the bandlimiting effect of the CBIR resampling at a lower rate than the design sample rate, according to one embodiment. As shown in FIG. 4A, the impulse response prototype 400 is a virtualization profile stored at the profile database 320 with a design sample rate of R_d. An audio input 401 has a sample rate of R_l. where $R_l < R_d$. The profile-processing module 330 first interpolates 405 the impulse response prototype 400 to obtain an CBIR 410, which is subsequently resampled 407 to produce an impulse response 420 with a target sample rate of R_l. Audio output 422 is the result of filtering the audio input 401 through the resampled impulse response 420. Obviously, the audio output 422 has a narrower bandwidth compared to the ideal audio output 412, which is filtered by the CBIR. This process demonstrates a finite impulse response (FIR) design technique based on sampling the continuous impulse response of the ideal filter represented by the interpolated CBIR.

[0062] Similarly, FIG. 4B is a diagram illustrating the bandlimiting effect of the CBIR resampling at a higher rate than the design sample rate, according to one embodiment. In FIG. 4B, audio input 402 has a sample rate of R_h. where $R_h > R_d$. The profile-processing module 330 resamples 409 the interpolated CBIR 410 to produce an impulse response 430 with a target sample rate of R_h. Audio output 432 is the result of filtering the audio input 402 through the resampled impulse response 430. The resulting audio output 422 is bandlimited at the original Nyquist frequency of the original impulse response. Alternatively, additional bandwidth extension techniques can be applied to resample impulse response 430, which may extend the audio bandwidth of the audio output 432 to that defined by R_h of the audio input 402. But this is generally unnecessary when a sufficient prototype 400 was captured.

[0063] Referring back to FIG. 3. In one embodiment, the measurement server 112 and/or the profile database 320 may have limited memory block or storage space for each virtualization profile (e.g. an impulse response prototype). For example, a profile or filter may be optimized for a fixed length of 1K or 1024 taps. On the other hand, filters are measured and sampled for a certain amount of time. At half of a design rate, there will be only half of the number of taps sampled, while at twice of the design rate, there will be twice of the number of taps recorded. It is therefore preferable to design the filter at the highest possible rate, which fits the fixed filter block length. If the filter is sampled at lower rates, the remaining memory space will be padded with zero taps.

[0064] Note that since they do not rely on feedback, FIR filters including those resampled through bandlimited interpolation are inherently stable and relatively tolerant of small errors in individual filter coefficients. Hence, the method is suitable for application at various resolutions and fixed-point implementation. However, like any operations, cumulative error could eventually accrue to audible noise in audio applications. As such it is preferable to always refer to the original filter coefficients stored in the profile database 320 for each interpolation and resample operation.

[0065] The bandlimited interpolation is suitable for resampling audio particularly because it reconstructs signal sampled at given points rather than approximating the signal through or around the sample points. In one embodiment, a moving *sinc* function is utilized for interpolating the impulse response prototype; the *sinc* function serves as the band-limiting low pass filter. In practice, since an ideal *sinc* does not exist to evaluate each sample to infinity, a special windowed *sinc* function is used instead. Generally this is achieved with a Kaiser window to control the trade-off between the stop-band attenuation and pass-band transition width. By combining the window and the *sinc* function, an efficient table implementation can be constructed, which is indexed by the relative time step between the sample rates.

[0066] In one embodiment, the profile-processing module 330 also provides controlled frequency scaling. For example, a scaling of a high-frequency resonance can render HRTFs a personalization effect roughly associated with ear-size and/or shape. To achieve this effect, the sample rate of a filter is adjusted by a factor relative to the true audio sampling rate. A factor of 1 (i.e. equal to the true audio rate) means no scaling. A factor less than 1 scales spectral features higher in frequency, while a factor greater than 1 scales spectral features lower in frequency. The frequency scaling reflects compression or expansion of the impulse response in time domain caused by the difference between the sample rates of the filter and the signal.

[0067] FIG. 5 is a block diagram illustrating functional modules within a headphone 162 for the cloud-based digital audio virtualization service. In one embodiment, the headphone 162 comprises a network interface 510, a profile database 520, a profile-processing module 530, and an audio processor 540. As used herein, the term "module" refers to a hardware and/or software unit used to provide one or more specified functionalities. Thus, a module can be implemented in hardware, software or firmware, or a combination of thereof. The headphone 162 is only one example of many user devices 160 (e.g., smartphone 163, laptop 166, personal audio player, A/V receiver, television, or any other device capable of playing audio and receiving user input), which may comprise different and/or fewer or more functional modules. In some embodiments, the headphone 162 may be coupled to another playback device, which include part or all of the modules described herein.

[0068] The headphone 162 communicates with the cloud server 120 via the network interface 510, which may be wired or wireless. The headphone 162 may be associated with a unique user account for the audio virtualization service at the cloud server 120. The user account may include information about the user of the headphone 162, such as the user's identification information, the user's hearing profiles and/or playback device profiles, and other user preferences. The network interface 510 forwards a user request 325 for virtualization profiles to the cloud server 120 and receives response 345 including one or more virtualization or room measurement profiles from the cloud server 120. The virtualization profiles received by the network interface 510 can be associated with the user account and passed to the profile memory 520 for use and storage. The virtualization profiles may be transmitted to the headphone 162 embedded in metadata of the audio content or separately from the audio content.

[0069] In case the virtualization profiles being acquired separately from the audio content, the headphone 162 may communicate with the cloud server 120 in advance or on the fly when the user attempts playback of some audio content to determine whether one or more virtualization profiles are associated or intended for the audio content. Hence, the virtualization profiles may be received prior to receiving the audio content, after receiving the audio content, or during reception of the audio content. Once the response 345 including one or more virtualization profiles is downloaded at the network interface 510, the profile-processing module 530 can process the virtualization profile, and the audio processor 540 applies the virtualization profile on the audio content. In one embodiment, the downloaded profiles may be stored in the profile memory 520 after the playback in case they are needed again later.

[0070] In some embodiments, the downloaded virtualization profile includes the resampled room measurement profiles, such as HRTF filter coefficients resampled to match the sample rate of the audio content. In this case, the interpolation and resampling of the virtualization profiles is performed by the cloud server 120 at the request of the headphone 162 indicating the target sample rate of the audio content. The profile memory 520 then passes the virtualization profiles to the audio processor 540, which processes the audio content by performing a direct convolution of the audio content with the downloaded virtualization profiles. In addition, if the virtualization profile includes early room response parameters and late reverberation parameters, then the profile-processing module 530 may create an acoustic model of the measurement room 110 and forward the acoustic model to the audio processor to process the audio content. In this case, the early room response parameters and the late reverberation parameters may be convolved with the audio content by the audio processor 540.

[0071] Alternatively, the headphone 162 or any other user devices may request original virtualization profiles, such as HRTF filter coefficients, at the design sample rates from the cloud server 120 without any processing. After the original virtualization profiles are received at the network interface 510 and stored at the profile memory 520, the profile-processing module 530 can process the virtualization profile locally. For example, if the design sample rates of the original virtualization profiles are different from the target sample rates of the audio content, the profile processing module 530 first needs to perform interpolation on the original HRTF and BRIR filters, to obtain a continuous bandlimit impulse response (CBIR). The interpolated CBIR is then resampled at the target sample rate before passing to the audio processor. The resampled filter coefficients can also be stored and/or cached by the profile memory 520 if necessary.

[0072] In some embodiments, the profile-processing module 530 and the audio processor 540 may process the virtualization profiles and the audio content at the time of playback and/or prior to the time of playback. Alternatively, in other embodiments, the processing of the audio content and the virtualization profiles may be distributed to other user devices. For example, the audio content may be pre-processed with some virtualization profiles at a local server and transmitted to headphone 162. Furthermore, the cloud-based virtualization may be constructed in such a way as to allow pre-processing of audio by content producers. This process may generate an optimized audio track designed to enhance user device playback in a manner specified by the content producer or to retain the desired attributes of the originally mixed surround soundtrack that provides the listener the sonic experience in the original studio.

[0073] The result of audio processing by the profile-processing module 530 and the audio processor 540 may be a bit stream that can be decoded using any audio decoder. The bit stream may include a flag that indicates whether or not the audio has been processed with the virtualization profiles. If the bit steam is played back using a legacy decoder that does not recognize the flag, the content may still be played, but without showing any indication on the virtualization of the audio content.

[0074] FIG. 6 is a detailed interaction diagram illustrating an example process for providing cloud-based digital audio virtualization, according to one embodiment. It should be noted that FIG. 6 only demonstrates one of many ways in which the embodiments of the cloud-based virtualization may be implemented. The method for providing the digital audio virtualization service involves the measurement server 112, the cloud server 120, and the user devices 160. The method begins with the measurement server 112 capturing 610 measurements in a mixing studio or a listening room with high fidelity audio equipment (e.g., from measurement room 110). Based on the room measurements, the measurement server 112 generates 612 virtualization profiles, such as head-related transfer function (HRTF) or binaural room impulse response (BRIR) filters, and then uploads 614 the virtualization profiles to the cloud server 120.

[0075] The cloud server 120 may process 620 the virtualization profiles uploaded by the measurement server 112. For example, the cloud server 120 may validate, encode, encrypt, and compress the virtualization profiles before storing 622 them in its database (e.g., profile database 320). The virtualization profiles, such as HRTF and BRIR filters, are often stored as filter coefficients sampled at a design sample rate. As described above, it is preferable to design the filter at the highest possible sample rate, which fits the filter storage block length for the purpose of interpolation and resampling. For example, a virtualization filter can be sampled as high as 192kHz with 64 bit A/D convertor length.

[0076] Assume that user devices 160 now request 630 for a virtualization profile for certain digital audio content with a specific sample rate, which may or may not be the same as the design sample rates of the virtualization profiles stored at the cloud server 120. Receiving the request, the cloud server 120 first determines 632 whether the requested sample rate equals to the design sample rate of the virtualization profile. If the requested sample rate is different from the design sample rate, the cloud server 120 can resample 634 the virtualization profile in response to the request. The resampling process may include, for example, interpolating the original HRTF or BRIR filters to obtain a continuous bandlimit impulse response (CBIR), and resampling the interpolated CBIR to match the target sample rate indicated by the request from the user devices 160. The cloud server 120 then transmits 636 the resampled virtualization profile with the requested sample rate to the user devices 160. If the requested sample rate is the same as the design sample rate, the cloud server 120 can simply transmits 636 the request virtualization profiles to the user devices 160 without resampling.

[0077] After the user devices 160 receive the response from the cloud server 160, the user devices 160 can filter 638 the digital audio content with the virtualization profile for a reproduction of the digital audio content simulating the production environment represented by the virtualization profile. For example, the user devices 160 may process the audio content by performing a direct convolution of the audio content with the downloaded virtualization filters of the profiles, so that the audio content is virtualized with similar effect of playback over the loudspeakers in the measurement room 110.

[0078] The user devices 160 may also directly request 640 the original virtualization profiles 640 or without specifying any sample rate from the cloud server 160, which will respond by transmitting 642 the requested virtualization profile with design sample rate (without any resampling). After receiving the response from the cloud server 160, the user devices 160 can resample 644 the virtualization filter of the profile to the required sample rate of the audio content before filtering 646 the audio content with at least the virtualization profile for a playback with environment virtualization effect. The resampling operation performed at the user devices 160, for example, may include a direct convolution of the audio content with the resampled virtualization filters of the profiles.

[0079] In conclusion, the method and apparatus disclosed in embodiments deliver a flexible filter design that adapts to variable sample rates for influencing the reproduction of the audio signal. The digital filters stored in the cloud servers are designed with the highest possible sample rate for the purpose of interpolation and resampling to different target sample rates. User devices, when plays back digital audio content, may directly download those digital filters resampled to proper sample rates by the cloud server, or acquire and resample the digital filters locally for virtualized listening experience.

[0080] The particulars shown herein are by way of example and for purposes of illustrative discussion of the embodiments of the present invention only, and are presented in the case of providing what is believed to be the most useful and readily understood description of the principles and conceptual aspects of the present invention. In this regard, no attempt is made to show particulars of the present invention in more detail than necessary for the fundamental understanding of the present invention, the description taken with the drawings make apparent to those skilled in the art how the several forms of the present invention may be embodied in practice.

* * *

CLAIMS

What is claimed is:

1. A method for processing an audio signal to influence the reproduction of the audio signal, comprising:

sending a request to a server computer for a virtualization profile, wherein the request specifies a requested sample rate for the virtualization profile, and wherein the virtualization profile defines a digital audio filter;

receiving from the server computer the virtualization profile with the requested sample rate; and

filtering the audio signal based on at least the virtualization profile by performing a convolution of the audio signal with the virtualization profile with the requested sample rate.

2. The method of claim 1, wherein the virtualization profile represents an acoustic model of a production environment.

3. The method of claim 1, further comprising causing the audio signal to be reproduced as sound through an audio transducer.

4. A method for processing an audio signal to influence the reproduction of the audio signal, comprising:

requesting a virtualization profile from a server computer, wherein the virtualization profile defines a digital audio filter;

receiving from the server computer the requested virtualization profile with a design sample rate;

resampling the virtualization profile at a required sample rate for the audio signal, responsive to a difference between the required sample rate and the design sample rate; and

filtering the audio signal based on at least the virtualization profile with the required sample rate.

5. The method of claim 4, wherein resampling the virtualization profile comprises:

interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and

resampling the CBIR at the required sample rate.

6. The method of claim 4, wherein filtering the audio signal comprises performing a convolution of the audio signal with the virtualization profile with the required sample rate.

7. The method of claim 4, further comprising causing the audio signal to be reproduced as sound through an audio transducer simulating a production environment.

8. A computer-implemented method for influencing reproductions of audio signals with virtualization profiles, the method comprising:

storing a virtualization profile with a design sample rate, wherein the virtualization profile defines a digital audio filter;

receiving a request for the virtualization profile from a client device, wherein the request specifies a requested sample rate for the virtualization profile;

resampling, by a computer processor, the stored virtualization profile at the requested sample rate, responsive to a difference between the requested sample rate and the design sample rate; and

transmitting the virtualization profile with the requested sample rate to the client device.

9. The method of claim 8, wherein the digital audio filter represents an acoustic model of a production environment comprising at least one of a finite impulse response (FIR) filter, an infinite impulse response (IIR) filter, and a feedback delay network (FDN) filter.

10. The method of claim 9, wherein the virtualization profile causes the audio signal to be reproduced through an audio transducer simulating the production environment.

11. The method of claim 8, wherein the virtualization profile is stored as a series of filter coefficients in fixed point or float point values.

12. The method of claim 8, wherein resampling the virtualization profile comprises:

interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and

resampling the CBIR at the requested sample rate.

13. The method of claim 12, wherein resampling the CBIR at a lower sample rate than the design sample rate results in fewer filter coefficients, and wherein resampling the CBIR at a higher sample rate than the design sample rate results in more filter coefficients.

14. The method of claim 8, further comprising scaling the virtualization profile to a different sample rate to achieve a subjective audio effect.

15. A non-transitory computer-readable storage medium storing computer-executable instructions that when executed cause one or more processors to perform operations comprising:

storing a virtualization profile with a design sample rate, wherein the virtualization profile defines a digital audio filter;

receiving a request for the virtualization profile from a client device, wherein the request specifies a requested sample rate for the virtualization profile;

resampling the stored virtualization profile at the requested sample rate, responsive to a difference between the requested sample rate and the design sample rate; and

transmitting the virtualization profile with the requested sample rate to the client device.

16. The non-transitory computer-readable storage medium of claim 15, wherein the digital audio filter represents an acoustic model of a production environment comprising at least one of a finite impulse response (FIR) filter, an infinite impulse response (IIR) filter, and a feedback delay network (FDN) filter.

17. The non-transitory computer-readable storage medium of claim 16, wherein the virtualization profile causes the audio signal to be reproduced through an audio transducer simulating the production environment.

18. The non-transitory computer-readable storage medium of claim 15, wherein

resampling the virtualization profile further comprises:

interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and

resampling the CBIR at the requested sample rate.

19. An audio device for processing an audio signal, comprising:

a communication interface configured for:

sending a request to a server computer for a virtualization profile, wherein the request specifies a requested sample rate for the virtualization profile, and wherein the virtualization profile defines a digital audio filter simulating a virtualized environment; and

receiving from the server computer the requested virtualization profile with the requested sample rate;

a storage device for storing the received virtualization profile; and

a processor in communication with the storage device and the communication interface, the processor programmed for:

filtering the audio signal based on at least the virtualization profile by performing a convolution of the audio signal with the virtualization profile with the requested sample rate.

20. An audio device for processing an audio signal, comprising:

a communication interface configured for:

requesting a virtualization profile from a server computer, wherein the virtualization profile defines a digital audio filter simulating a virtualized environment; and

receiving from the server computer the requested virtualization profile with a design sample rate;

a storage device for storing the received virtualization profile; and

a processor in communication with the storage device and the communication interface, the processor programmed for:

resampling the virtualization profile at a required sample rate for the audio signal, responsive to a difference between the required sample rate and the design sample rate; and filtering the audio signal based on at least the virtualization profile with the required sample rate.

21. The audio device of claim 20, wherein resampling the virtualization profile comprises:

interpolating the virtualization profile to obtain a representation of continuous-time bandlimited impulse response (CBIR); and

resampling the CBIR at the required sample rate.

22. The audio device of claim 20, wherein filtering the audio signal comprises performing a convolution of the audio signal with the virtualization profile at the required sample rate.

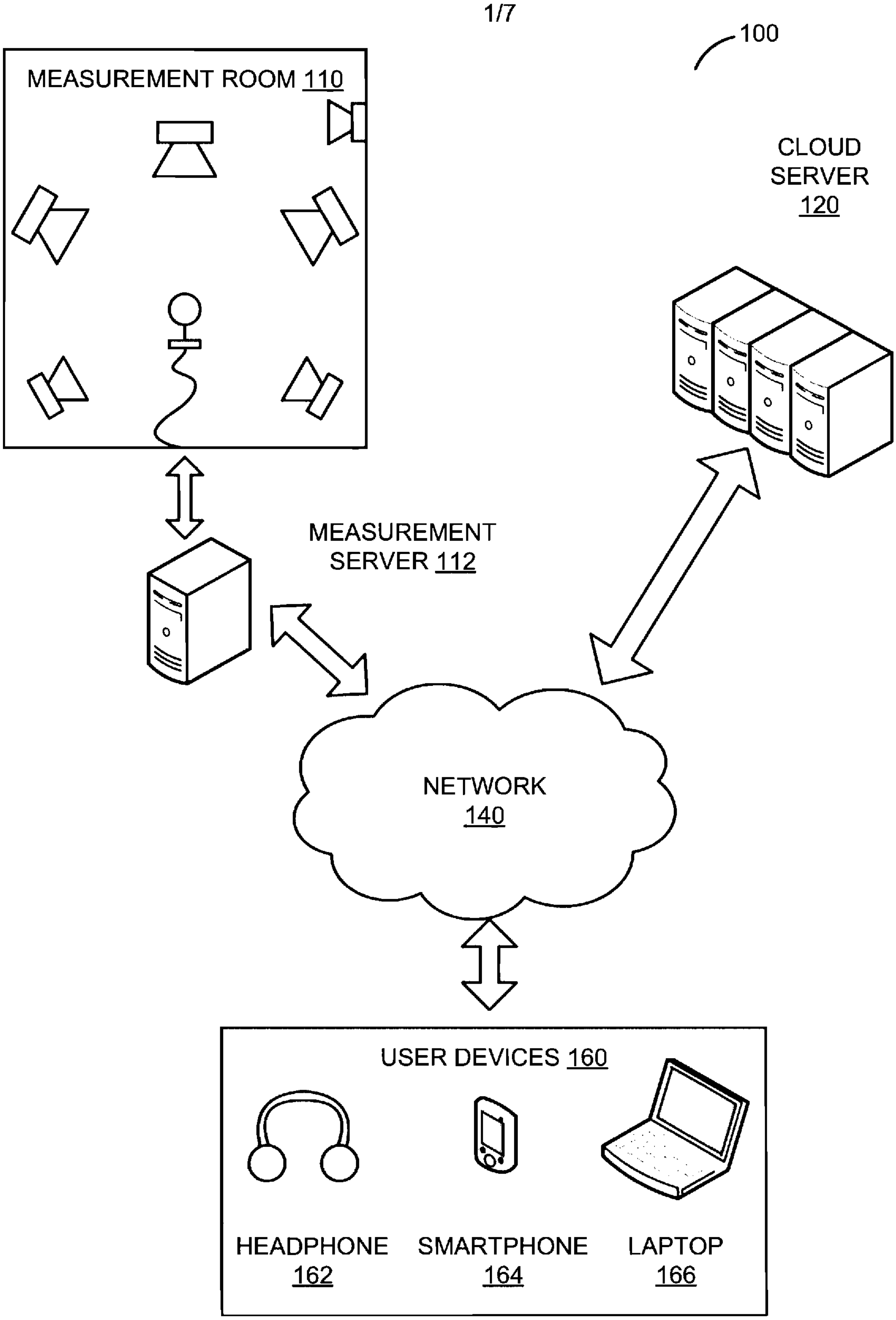

100
MEASUREMENT ROOM 110
CLOUD SERVER 120
MEASUREMENT SERVER 112
NETWORK 140
USER DEVICES 160
HEADPHONE 162
SMARTPHONE 164
LAPTOP 166

FIG. 1

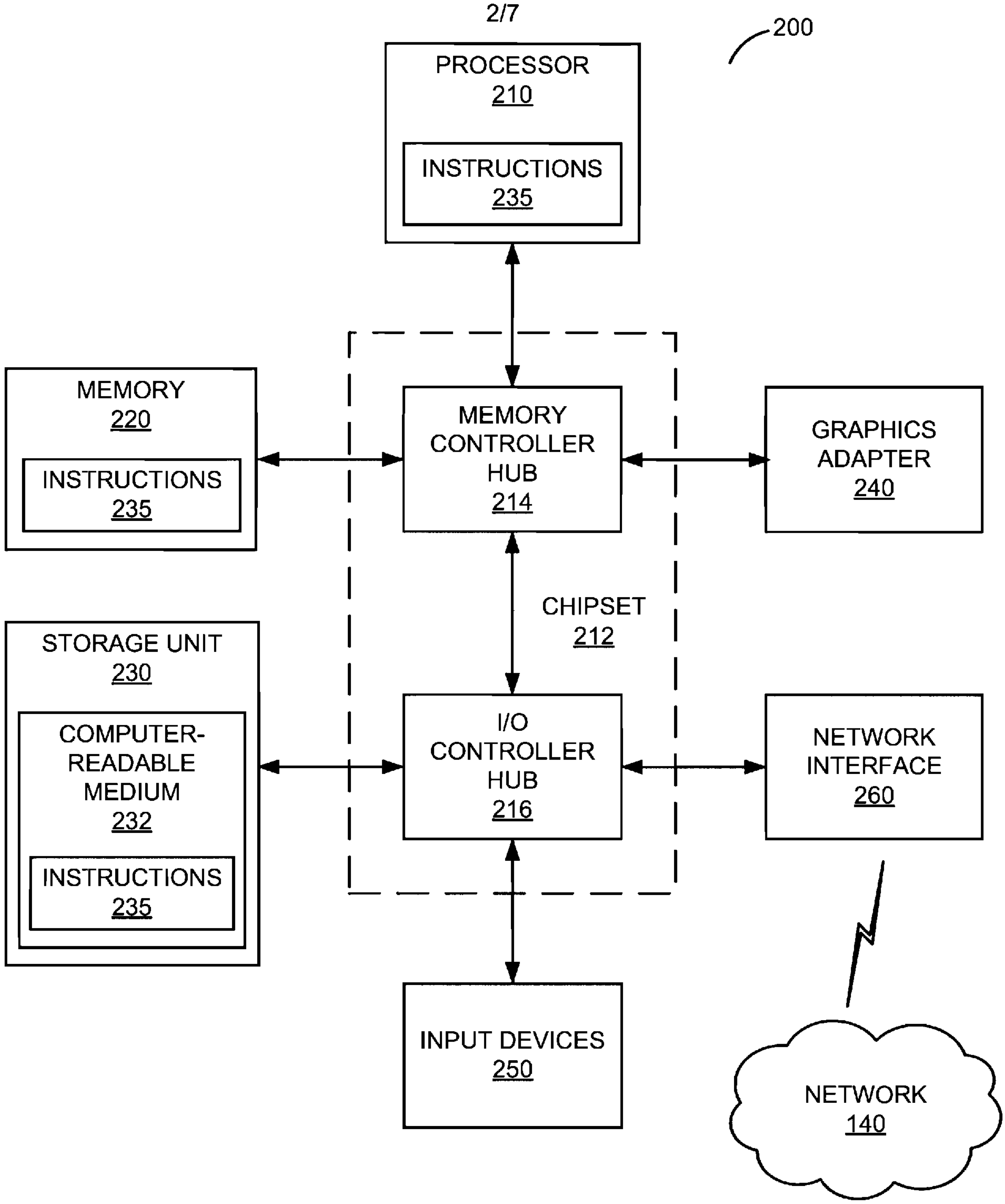

200
PROCESSOR
210
INSTRUCTIONS
235
MEMORY
220
INSTRUCTIONS
235
MEMORY CONTROLLER HUB
214
GRAPHICS ADAPTER
240
CHIPSET
212
STORAGE UNIT
230
COMPUTER-READABLE MEDIUM
232
INSTRUCTIONS
235
I/O CONTROLLER HUB
216
NETWORK INTERFACE
260
INPUT DEVICES
250
NETWORK
140

FIG. 2

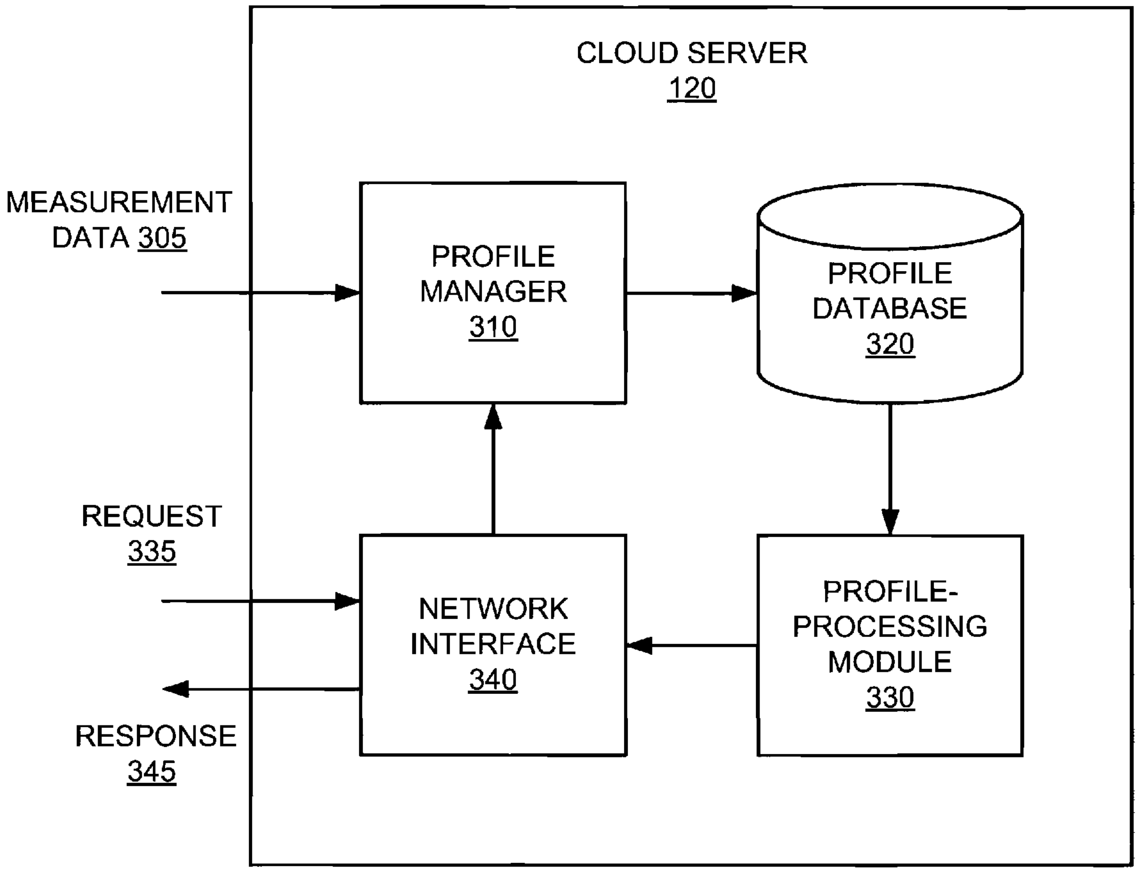

CLOUD SERVER
120
MEASUREMENT
DATA 305
PROFILE
MANAGER
310
PROFILE
DATABASE
320
REQUEST
335
NETWORK
INTERFACE
340
PROFILE-
PROCESSING
MODULE
330
RESPONSE
345

FIG. 3

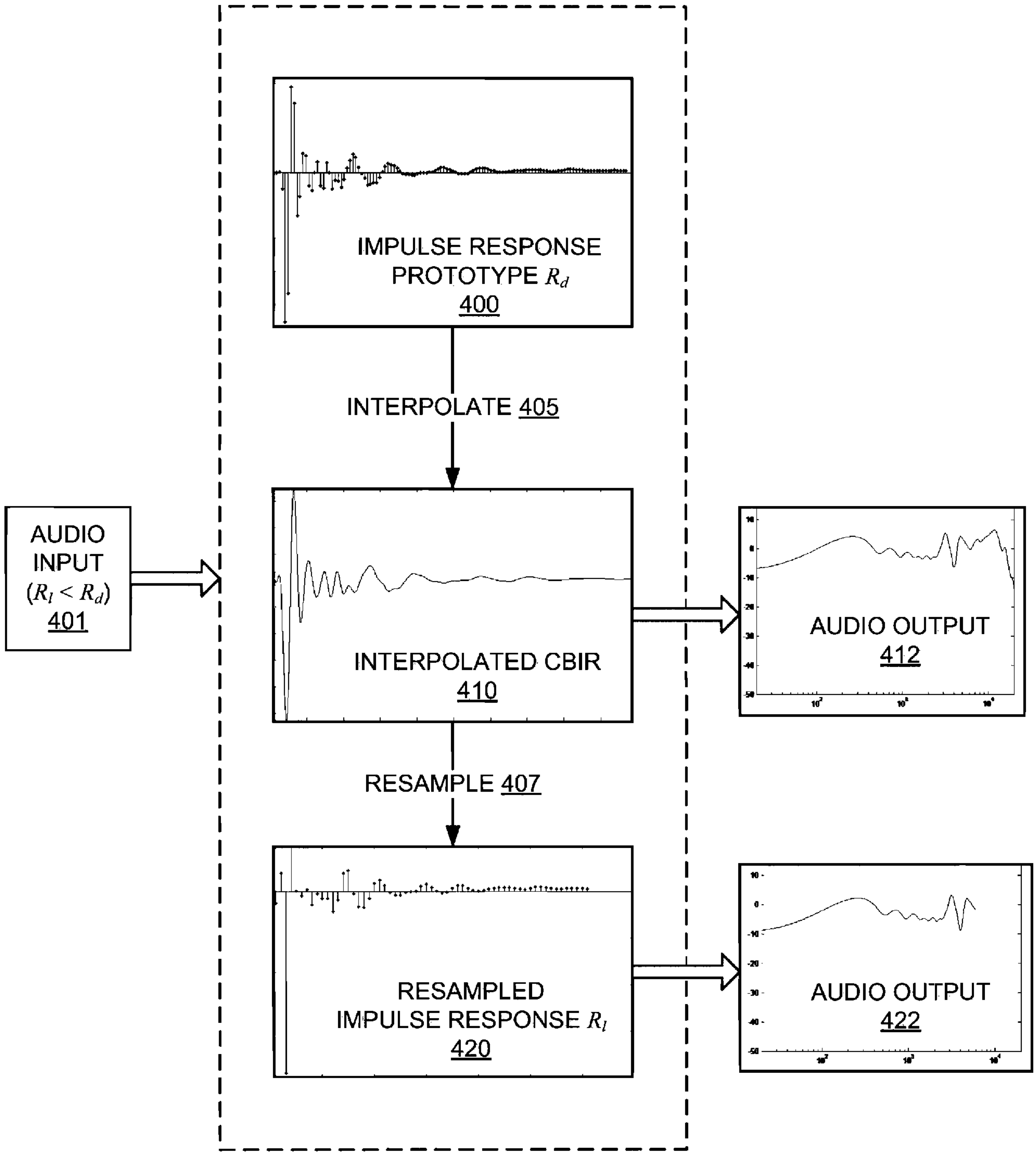
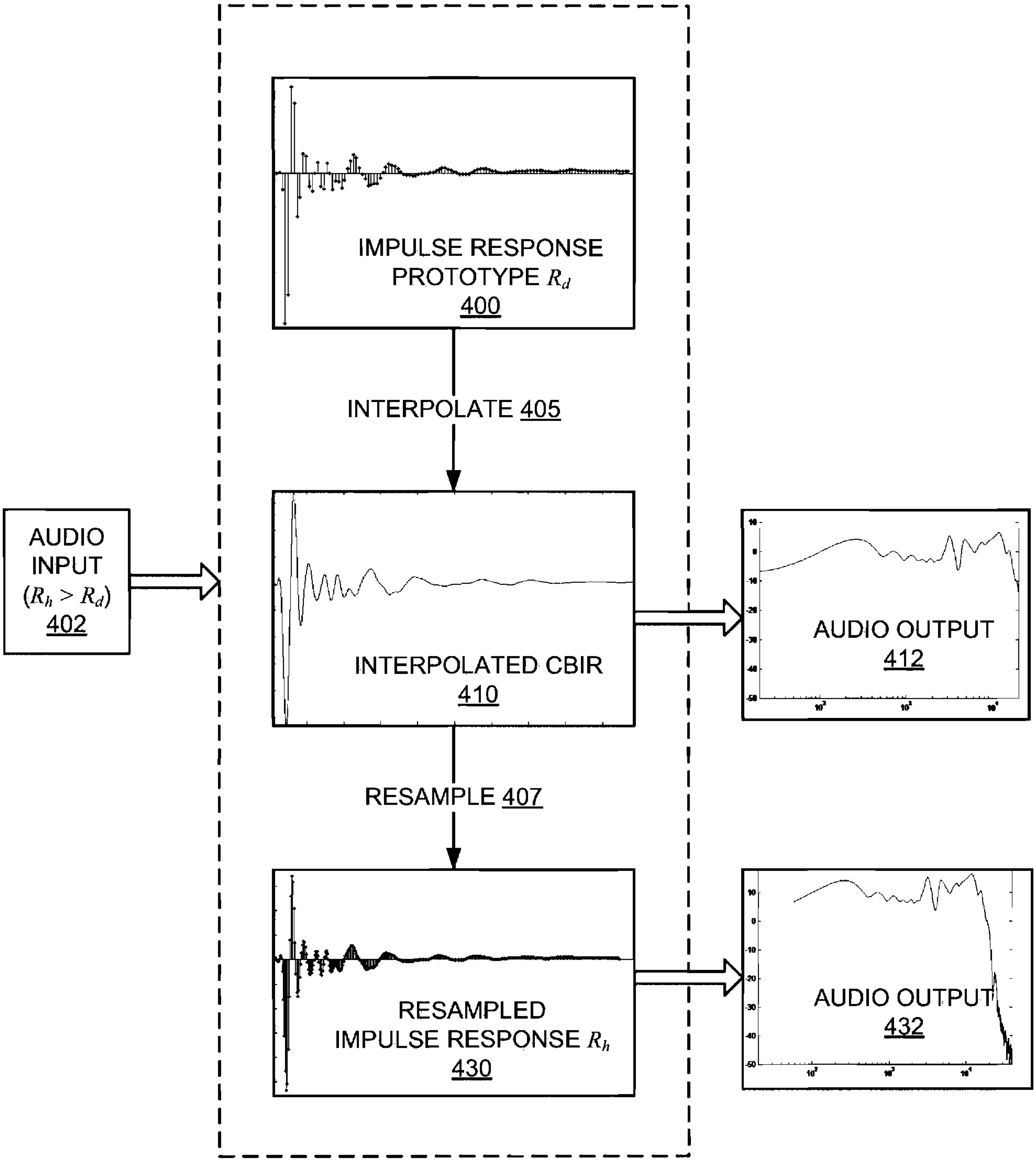

IMPULSE RESPONSE PROTOTYPE R_d
400
INTERPOLATE 405
AUDIO INPUT
$(R_l < R_d)$
401
INTERPOLATED CBIR
410
AUDIO OUTPUT
412
RESAMPLE 407
RESAMPLED IMPULSE RESPONSE R_l
420
AUDIO OUTPUT
422

FIG. 4A



IMPULSE RESPONSE PROTOTYPE R_d 400
INTERPOLATE 405
AUDIO INPUT ($R_h > R_d$) 402
INTERPOLATED CBIR 410
AUDIO OUTPUT 412
RESAMPLE 407
RESAMPLED IMPULSE RESPONSE R_h 430
AUDIO OUTPUT 432

FIG. 4B

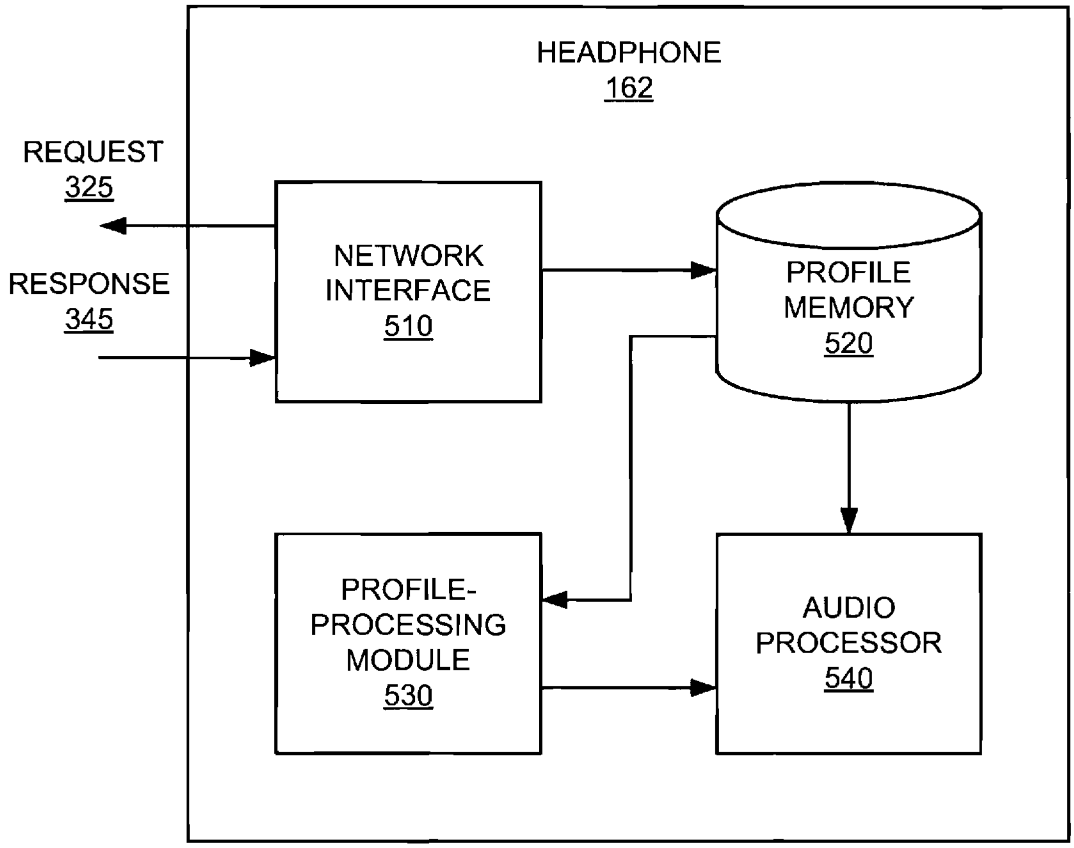

HEADPHONE
162
REQUEST
325
RESPONSE
345
NETWORK
INTERFACE
510
PROFILE
MEMORY
520
PROFILE-
PROCESSING
MODULE
530
AUDIO
PROCESSOR
540

FIG. 5

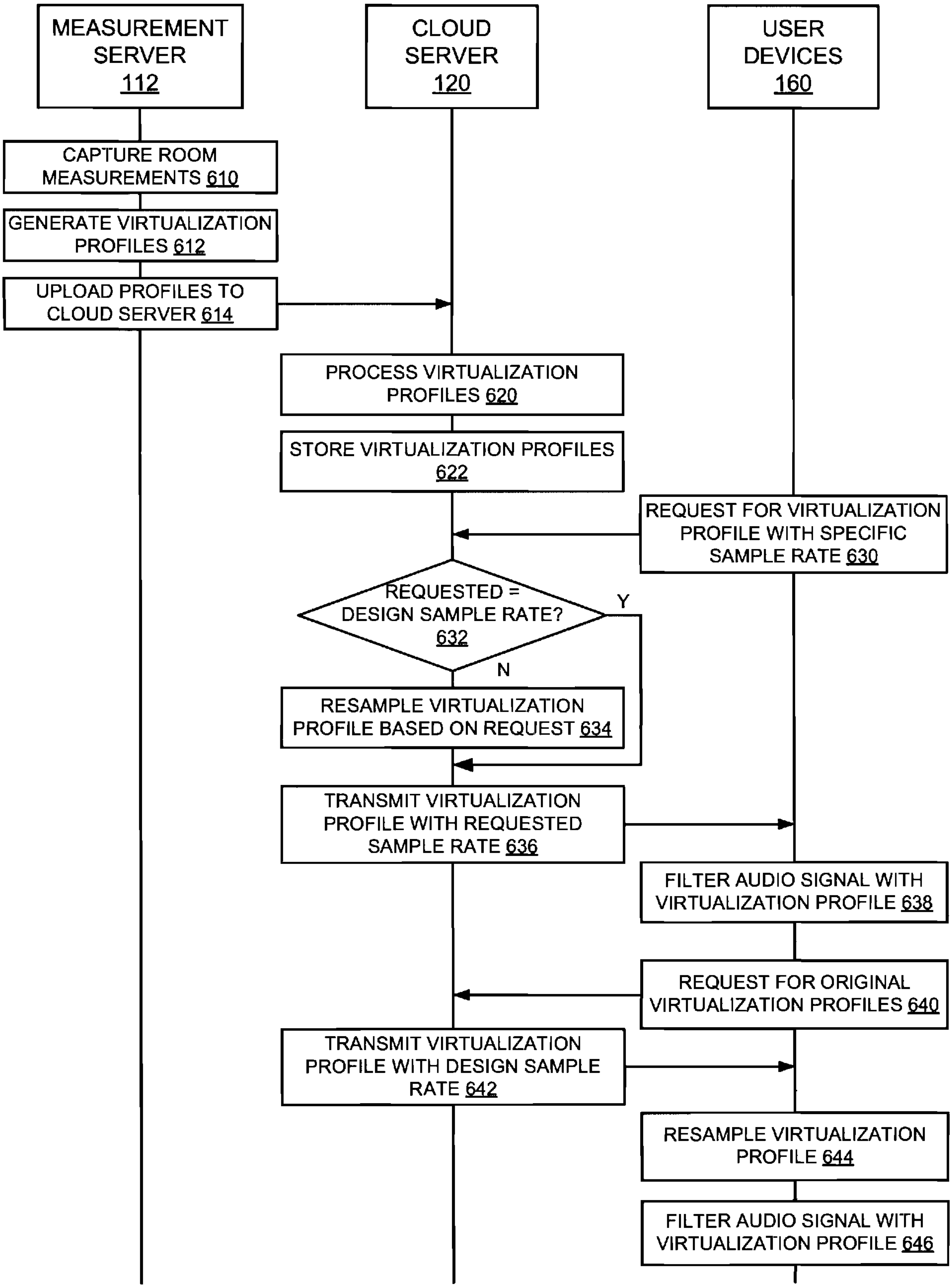

MEASUREMENT SERVER 112
CLOUD SERVER 120
USER DEVICES 160
CAPTURE ROOM MEASUREMENTS 610
GENERATE VIRTUALIZATION PROFILES 612
UPLOAD PROFILES TO CLOUD SERVER 614
PROCESS VIRTUALIZATION PROFILES 620
STORE VIRTUALIZATION PROFILES 622
REQUEST FOR VIRTUALIZATION PROFILE WITH SPECIFIC SAMPLE RATE 630
REQUESTED = DESIGN SAMPLE RATE? 632
Y
N
RESAMPLE VIRTUALIZATION PROFILE BASED ON REQUEST 634
TRANSMIT VIRTUALIZATION PROFILE WITH REQUESTED SAMPLE RATE 636
FILTER AUDIO SIGNAL WITH VIRTUALIZATION PROFILE 638
REQUEST FOR ORIGINAL VIRTUALIZATION PROFILES 640
TRANSMIT VIRTUALIZATION PROFILE WITH DESIGN SAMPLE RATE 642
RESAMPLE VIRTUALIZATION PROFILE 644
FILTER AUDIO SIGNAL WITH VIRTUALIZATION PROFILE 646

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US15/38635

A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - G06F 17/17, 17/10; H04R 29/00 (2015.01)
CPC - H04S 7/306, 7/304, 7/303
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC(8): H04R 29/00; G06F 17/17, 17/10 (2015.01)
CPC: H04S 2420/01, 7/306, 7/304, 7/303, 7/302, 7/30

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
PatSeer (US, EP, WO, JP, DE, GB, CN, FR, KR, ES, AU, IN, CA, INPADOC Data); Google Patent, Google Scholar, IEEE, ProQuest
Search terms used: virtualization, profile, sample. rate, audio, server, user, client, request, digital, filter, FIR, convolution, transducer, processor, computer, continuous, time, bandlimited, impulse, response, CBIR, lower

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2005/0248476 A1 (WISER, P et al.) November 10, 2005; paragraphs [0040], [0041], [0043], [0044], [0046], [0047], [0049], [0071], [0075]	1-12, 14-22
Y	WO 2013/049125 A1 (ACTIWAVE AB) April 4, 2013; paragraphs [00142] & [00143], equation 2	1-3, 6, 19, 22
Y --- A	US 2006/0212503 A1 (BECKMANN, P et al.) September 21, 2006; figure 7, paragraphs [0026] & [0027]	4-12, 14-18, 20-22 --- 13
Y	US 2006/0045294 A1 (SMYTH, S) March 2, 2006; paragraphs [0134], [0140], [0174], claims 1 & 4	11, 14

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
02 November 2015 (02.11.2015)	04 DEC 2015

Name and mailing address of the ISA/	Authorized officer
Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-8300	Shane Thomas PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US15/38635

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:
See supplemental sheet.

1. ☒ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☒ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/US15/38635

-***-Continued from Box III-***-
This application contains the following inventions or groups of inventions which are not so linked as to form a single general inventive concept under PCT Rule 13.1. In order for all inventions to be examined, the appropriate additional examination fee must be paid.

Group I: Claims 1-3 & 19 appear to be directed towards filtering the audio signal based on at least the virtualization profile by performing a convolution of the audio signal with the virtualization profile with the requested sample rate.

Group II: Claims 4-18 & 20-22 appear to be directed towards resampling, by a computer processor, the stored virtualization profile at the requested sample rate, responsive to a difference between the requested sample rate and the design sample rate.
The inventions listed as Groups I & II do not relate to a single general inventive concept under PCT Rule 13.1 because, under PCT Rule 13.2, they lack the same or corresponding special technical features.

The special technical features present in Group I are at least filtering the audio signal based on at least the virtualization profile by performing a convolution of the audio signal with the virtualization profile with the requested sample rate, and wherein the virtualization profile defines a digital audio filter simulating a virtualized environment, which Group II does not have. Group II has at least resampling the virtualization profile at a required sample rate for the audio signal, responsive to a difference between the required sample rate and the design sample rate, which Group I does not have.

The common technical features of Groups I & II are at least a method for processing an audio signal to influence the reproduction of the audio signal, comprising: sending a request to a server computer for a virtualization profile, wherein the request specifies a requested sample rate for the virtualization profile, and wherein the virtualization profile defines a digital audio filter; receiving from the server computer the virtualization profile with the requested sample rate and filtering the audio signal based on at least the virtualization profile with the required sample rate.These common features are previously disclosed by US 2005/0248476 A1 to Wiser, P et al. (hereinafter 'Wiser'). Wiser discloses a method for processing an audio signal to influence the reproduction of the audio signal (a method for processing an audio file 104 corresponding to an audio signal 106 to influence the reproduction, whether it is mono-aural, stereo, etc., of the encoded audio signal; paragraphs [0044] & [0071]), comprising: sending a request to a server computer for a virtualization profile (a request is sent, based on a user selection, to a remote audio profile database 110 via a single-channel ISDN (server) for a audio processing profile 304A (virtualization profile); paragraphs [0040] & [0071]), wherein the request specifies a requested sample rate for the virtualization profile (the profile request includes a sample rate field 404A for the audio processing profile 304A; paragraphs [0043] & [0075]), and wherein the virtualization profile defines a digital audio filter (the audio processing profile 304A defines a digital audio filter via decimation and interpolation; paragraphs [0046] & [0047]); receiving from the server computer the virtualization profile with the requested sample rate (receiving the audio processing profile 304A from the server with the requested audio processing profile 304A which contains the sample rate field 404A; paragraph [0041]) and filtering the audio signal based on at least the virtualization profile with the required sample rate (the audio signal 106 is then filtered based on the audio processing profile 304A that contains the sample rate field 404A; Figures 1, 3, 4; paragraphs [0039], [0047], [0049], and [0059]).

Since the common technical feature is previously disclosed by Wiser these common features are not special and so Groups I & II lack unity.