US 20150281308A1

(19) United States

(12) Patent Application Publication

LIAO

(10) Pub. No.: US 2015/0281308 A1

(43) Pub. Date: Oct. 1, 2015

(54) METHOD AND SYSTEM FOR DOWNLOADING AND PLAYING MEDIA FILE, CLIENT, SERVER, AND STORAGE MEDIUM

(71) Applicant: **Tencent Technology (Shenzhen) Company Limitead**, Shenzhen (CN)

(72) Inventor: **Haizhen LIAO**, Shenzhen (CN)

(21) Appl. No.: **14/736,122**

(22) Filed: **Jun. 10, 2015**

Related U.S. Application Data

(63) Continuation of application No. PCT/CN2013/084384, filed on Sep. 27, 2013.

(30) **Foreign Application Priority Data**

Dec. 13, 2012 (CN) 201210538686.0

Publication Classification

(51) **Int. Cl.**

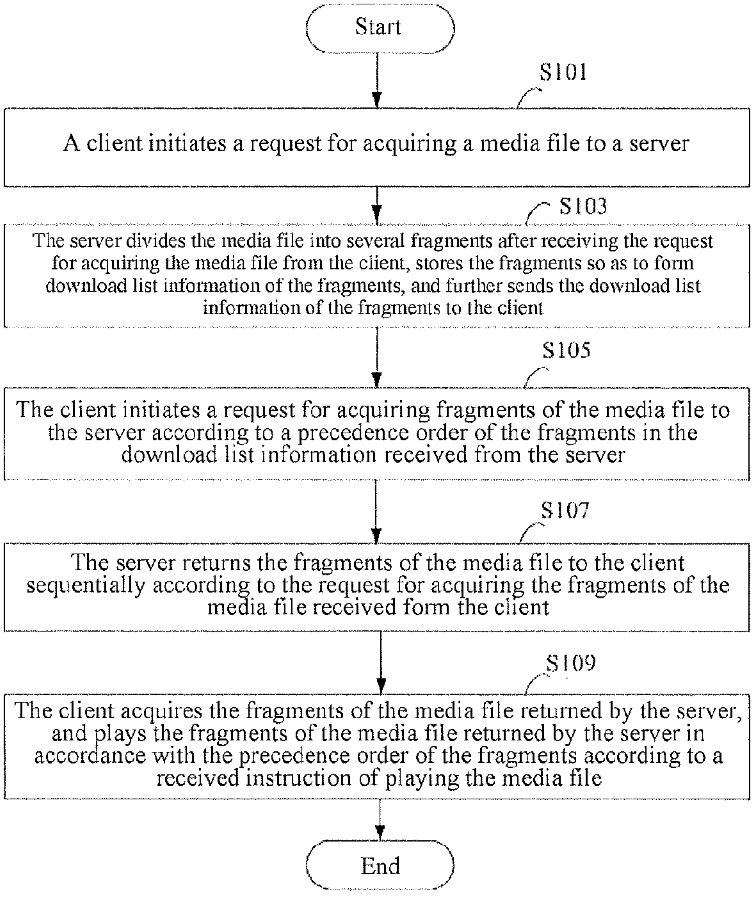
H04L 29/06	(2006.01)
H04L 29/08	(2006.01)

(52) **U.S. Cl.**

CPC ***H04L 65/604*** (2013.01); ***H04L 67/42*** (2013.01); ***H04L 67/10*** (2013.01)

(57) ABSTRACT

The present disclosure relates to method and system for downloading and playing a media file, a client, a server, and a storage medium. The client is configured to initiate a request for acquiring a media file to the server, the request including current network connection information of the client, receive download list information from the server, the download list information identifying an ordered list of fragments extracted from the media file by the server in accordance with the current network connection information of the client, initiate a request for acquiring fragments of the media file to the server, acquire the fragments of the media file returned by the server, and play the fragments of the media file according to the order of the fragments when receiving an instruction of playing the media file. The present disclosure can simply and quickly play a media file that is not downloaded completely.


Start
S101
A client initiates a request for acquiring a media file to a server
S103
The server divides the media file into several fragments after receiving the request for acquiring the media file from the client, stores the fragments so as to form download list information of the fragments, and further sends the download list information of the fragments to the client
S105
The client initiates a request for acquiring fragments of the media file to the server according to a precedence order of the fragments in the download list information received from the server
S107
The server returns the fragments of the media file to the client sequentially according to the request for acquiring the fragments of the media file received form the client
S109
The client acquires the fragments of the media file returned by the server, and plays the fragments of the media file returned by the server in accordance with the precedence order of the fragments according to a received instruction of playing the media file
End

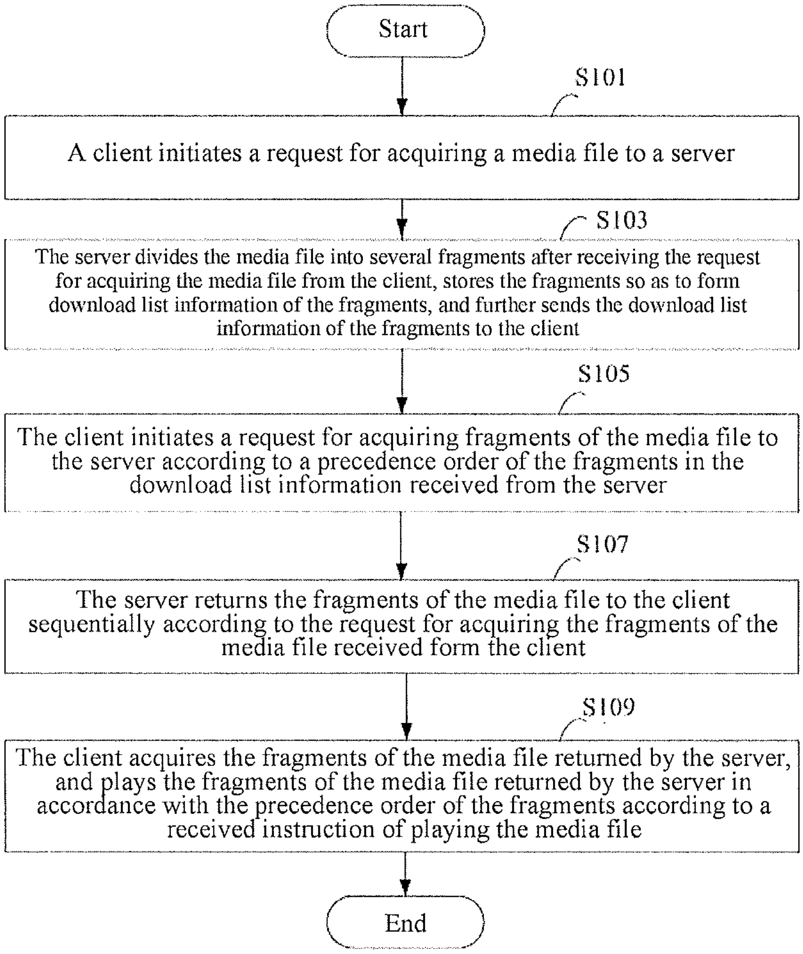

Start
S101
A client initiates a request for acquiring a media file to a server
S103
The server divides the media file into several fragments after receiving the request for acquiring the media file from the client, stores the fragments so as to form download list information of the fragments, and further sends the download list information of the fragments to the client
S105
The client initiates a request for acquiring fragments of the media file to the server according to a precedence order of the fragments in the download list information received from the server
S107
The server returns the fragments of the media file to the client sequentially according to the request for acquiring the fragments of the media file received form the client
S109
The client acquires the fragments of the media file returned by the server, and plays the fragments of the media file returned by the server in accordance with the precedence order of the fragments according to a received instruction of playing the media file
End

FIG. 1

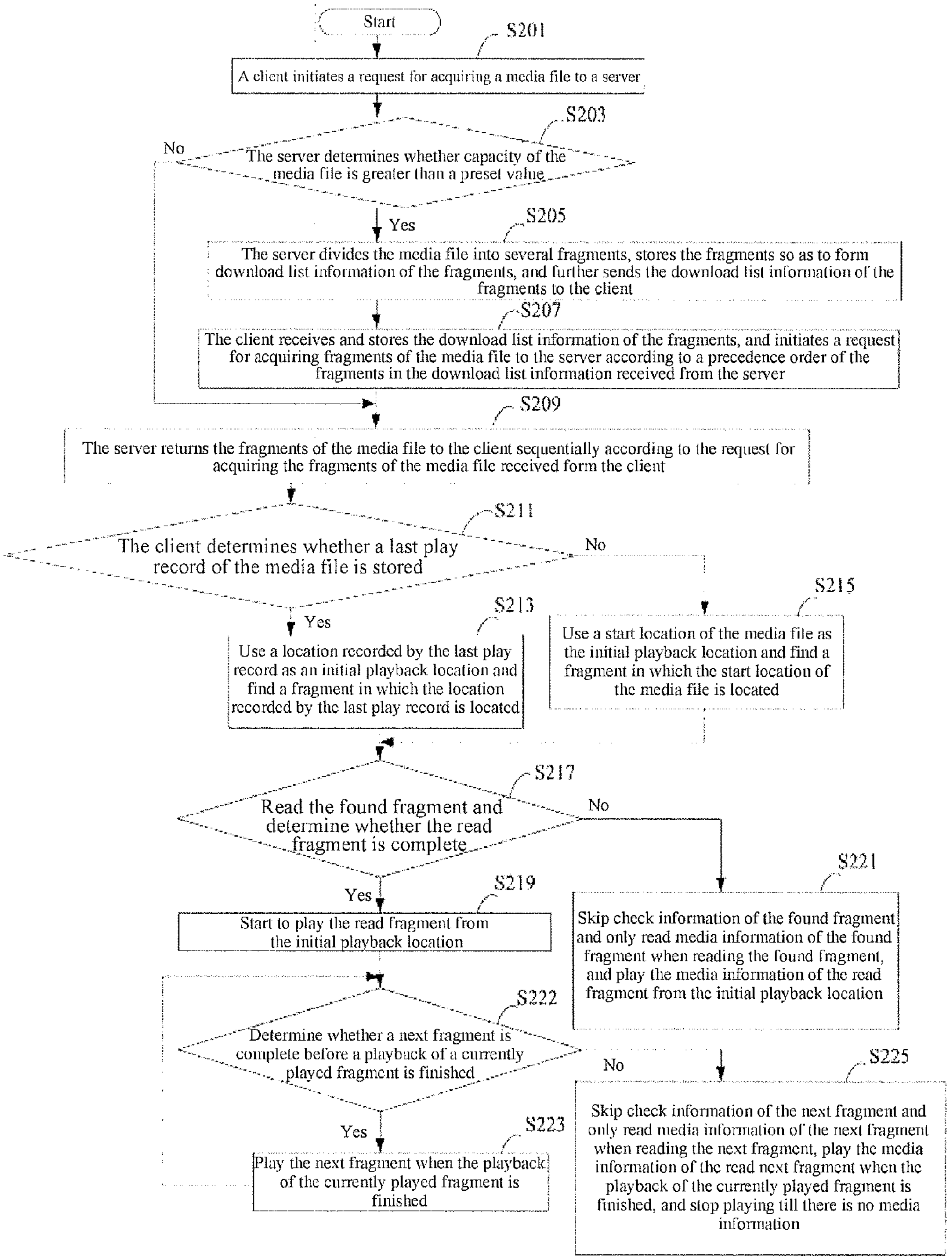

Start
S201
A client initiates a request for acquiring a media file to a server
S203
No
The server determines whether capacity of the media file is greater than a preset value
Yes
S205
The server divides the media file into several fragments, stores the fragments so as to form download list information of the fragments, and further sends the download list information of the fragments to the client
S207
The client receives and stores the download list information of the fragments, and initiates a request for acquiring fragments of the media file to the server according to a precedence order of the fragments in the download list information received from the server
S209
The server returns the fragments of the media file to the client sequentially according to the request for acquiring the fragments of the media file received form the client
S211
The client determines whether a last play record of the media file is stored
No
S215
Use a start location of the media file as the initial playback location and find a fragment in which the start location of the media file is located
Yes
S213
Use a location recorded by the last play record as an initial playback location and find a fragment in which the location recorded by the last play record is located
S217
Read the found fragment and determine whether the read fragment is complete
No
S221
Skip check information of the found fragment and only read media information of the found fragment when reading the found fragment, and play the media information of the read fragment from the initial playback location
Yes
S219
Start to play the read fragment from the initial playback location
S222
Determine whether a next fragment is complete before a playback of a currently played fragment is finished
No
S225
Skip check information of the next fragment and only read media information of the next fragment when reading the next fragment, play the media information of the read next fragment when the playback of the currently played fragment is finished, and stop playing till there is no media information
Yes
S223
Play the next fragment when the playback of the currently played fragment is finished

FIG. 2

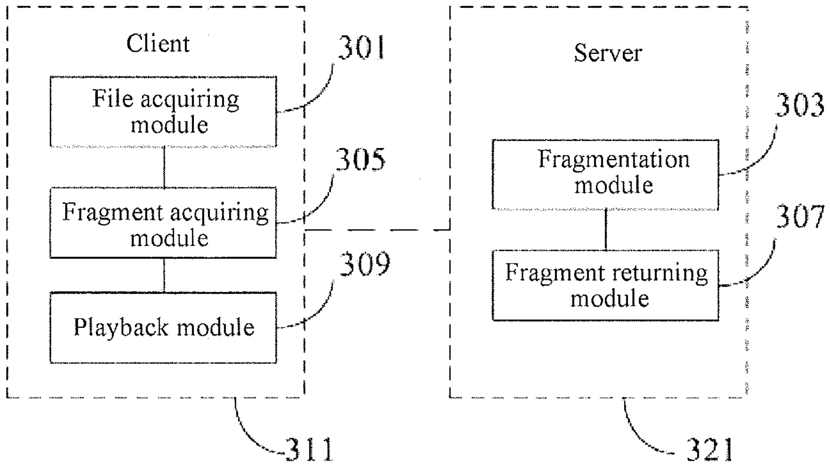

Client
File acquiring module
301
Fragment acquiring module
305
Playback module
309
311
Server
Fragmentation module
303
Fragment returning module
307
321

FIG. 3

METHOD AND SYSTEM FOR DOWNLOADING AND PLAYING MEDIA FILE, CLIENT, SERVER, AND STORAGE MEDIUM

RELATED APPLICATIONS

[0001] This patent application is a continuation application of PCT Patent Application No. PCT/CN2013/084384, entitled "METHOD AND SYSTEM FOR DOWNLOADING AND PLAYING MEDIA FILE, CLIENT, SERVER, AND STORAGE MEDIUM" filed on Sep. 27, 2013, which claims priority to Chinese Patent Application No. 201210538686.0, filed on Dec. 13, 2012, and entitled "METHOD AND SYSTEM FOR DOWNLOADING AND PLAYING MEDIA FILE", both of which are incorporated by reference in their entirety.

FIELD OF THE TECHNOLOGY

[0002] The present disclosure relates to the field of network information transmission technologies, and in particular, to a method and system for downloading and playing a media file, a client, a server, and a storage medium.

BACKGROUND OF THE DISCLOSURE

[0003] At present, with rapid development of network technologies, various media files such as digital music, video files are quite popular broadcast information on the Internet. These media files are generally stored in websites and broadcasted via a network, and therefore, a user needs to find a needed media file from a website and then directly download the media file if the user wants to download the needed media file locally, so as to facilitate appreciating play content of the media file locally.

[0004] However, in the downloading method, if the media file is not downloaded completely due to network reasons or other reasons, the media file that is not downloaded completely cannot be played locally. If the user wants to play the media file, the user needs to wait till the media file is downloaded completely, causing that user playback experiences of the media file are greatly lowered, and excellent feelings of the user during entertainment cannot be ensured.

SUMMARY

[0005] Therefore, the present disclosure provides method and system for downloading and playing a media file, a client, a server, and a storage medium, which can simply and quickly play a media file that is not downloaded completely.

[0006] Specifically, embodiments of the present application provide a method for downloading and playing a media file, running on a client and including the following steps: initiating a request for acquiring a media file to a server, the request including current network connection information of the client; receiving download list information from the server, wherein the download list information identifies an ordered list of fragments extracted from the media file by the server in accordance with the current network connection information of the client; initiating a request for acquiring the fragments of the media file to the server according to the order of the fragments in the download list information, and acquiring the fragments of the media file returned by the server; and playing the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

[0007] The embodiments of the present application further provide a method for downloading and playing a media file, running on a server and including the following steps: receiving a request for acquiring a media file from a client, the request including current network connection information of the client; dividing the media file into an ordered list of fragments according to the current network connection information of the client, storing the fragments so as to form download list information of the fragments, and further sending the download list information of the fragments to the client; receiving a request for acquiring the fragments of the media file from the client; and returning the fragments of the media file to the client sequentially according to the order of the fragments in the download list information.

[0008] According to the method provided in the embodiments of the present application, the embodiments of the present application further provide a client including one or more processors; memory; and a plurality of program modules, wherein the plurality of program modules, when executed by the one or more processors, cause the client to perform predefined operations, the plurality of program modules further including:

[0009] a file acquiring module, configured to initiate a request for acquiring a media file to a server, the request including current network connection information of the client, and receive download list information from the server, wherein the download list information identifies an ordered list of fragments extracted from the media file by the server in accordance with the current network connection information of the client;

[0010] a fragment acquiring module, configured to initiate a request for acquiring the fragments of the media file to the server according to the order of the fragments in the download list information, and acquire the fragments of the media file returned by the server; and

[0011] a playback module, configured to play the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

[0012] It may be known from the foregoing descriptions that in the present disclosure, the server divides the media file into an ordered list of fragments according to the current network connection information of the client and stores the fragments so as to form the download list information of the fragments; and the client initiates a request for acquiring fragments of the media file to the server according to an order of the fragments in download list information, and plays the fragments of the media file according to the order of the fragments when receiving an instruction of playing the media file. Because an optimization downloading method of dividing the media file and a method of playing the fragments are used, the downloading method is optimized, the media file may also be played according to an order of the fragments. Even if when other fragments of the media file are not downloaded completely, the downloaded fragments may also be normally played. In this way, a user may play a part of the media file that is downloaded completely at any time, which achieves an objective of simply and quickly playing a media file that is not downloaded completely.

[0013] The foregoing description merely briefly describes the technical solution of the present disclosure, so as to understand the technical solution of the present disclosure more clearly, and implement the present disclosure according to the content of the specification. Moreover, to make the foregoing and other objectives, characteristics, and advantages of the present disclosure more comprehensible, detailed descriptions are made in the following through preferred embodiments with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] FIG. **1** is a flowchart of steps of a method for downloading and playing a media file according to an embodiment of the present application;

[0015] FIG. **2** is a flowchart of steps of a method for downloading and playing a media file according to another embodiment of the present application; and

[0016] FIG. **3** is a block diagram of main architecture of a system for downloading and playing a media file according to an embodiment of the present application.

DESCRIPTION OF EMBODIMENTS

[0017] To further explain the technical means used in the present disclosure for achieving the intended objectives and the efficacy thereof, specific implementation manners, structures, features, and efficacy of a method and system for downloading and playing a media file, a client, a server, and a storage medium provided in the present disclosure are described in detail below with reference to the accompanying drawings and preferred embodiments.

[0018] The foregoing and other technical content, features, and efficacy of the present disclosure will be clearly presented in the detailed description of the preferred embodiments with reference to the drawings. Through the description of the specific implementation manners, the technical means used in the present disclosure for achieving the intended objectives and the efficacy can be understood more deeply and specifically. However, the accompanying drawings are just used for reference and description and not intended to limit the present disclosure.

[0019] FIG. **1** is a flowchart of steps of a method for downloading and playing a media file according to an embodiment of the present application. Referring to FIG. **1**, the method for downloading and playing a media file in this embodiment of the present application may include the following steps S**101**-S**109**:

[0020] Step S**101**: A client initiates a request for acquiring a media file to a server, the request including current network connection information of the client.

[0021] In this step, the media file may be a file such as music or a video. The client may be a mobile client such as a mobile phone. In some embodiments, the current network connection information of the client indicates whether the client has a wired or wireless connection to the server. For wireless connection, the connection information further indicates whether the connection is Wi-Fi or cellular network connection. Different types of network connection have different network transmission speeds, which may be used for determining the size of a fragment of a media file by the server.

[0022] Step S**103**: The server divides a media file into an ordered list of fragments according to the current network connection information of the client after receiving the request for acquiring the media file from the client, stores the fragments so as to form download list information of the fragments, and further sends the download list information of the fragments to the client.

[0023] In this step, a method of dividing the media file into the several fragments may be a method of performing average division according to a file size of the media file, or any other dividing method. If the average dividing method is used, the method may be that: if a media file that is 128 MB is divided into two fragments, the file size of each fragment is 64 MB. As noted above, the server may use the current network connection information when fragmenting the media file. For example, if the network transmission speed is below a preset value, the server may divide the media file into more fragments of smaller size such that it is more likely for the client to download a complete fragment within a short time period when the client has connection to the server. If the network transmission speed exceeds a preset value, the server may divide the media file into fewer fragments of larger size such that it is more likely for the client to download a large, complete fragment within a predefined time period when the client has connection to the server.

[0024] The download list information of each of the fragments includes information such as a uniform/universal resource locator (URL) stored in the fragment.

[0025] Step S**105**: The client initiates a request for acquiring fragments of the media file to the server according to an order of the fragments in the download list information received from the server.

[0026] Step S**107**: The server returns the fragments of the media file to the client sequentially according to the request for acquiring the fragments of the media file received from the client.

[0027] In this step, the order of the fragments may be a chronological order of the fragments (i.e., a chronological order in which the media file is played). For example, a fragment in which a start location of the media file is located is the first fragment; the fragments are sequentially arranged according to a play chronological order; and other fragments are sequentially the second fragment, the third fragment, and the like.

[0028] Step S**109**: The client acquires the fragments of the media file returned by the server, and plays the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

[0029] In this step, because the fragments are mutually independent, that is, the fragments may be independently played, fragments that are downloaded completely may also be independently played when other fragments are not downloaded completely.

[0030] In this embodiment of the present application and in the present disclosure, the server divides the media file into an ordered list of fragments according to the current network connection information of the client and stores the fragments so as to form the download list information of the fragments; and the client initiates a request for acquiring fragments of the media file to the server according to an order of the fragments in download list information, and plays the fragments of the media file according to the order of the fragments when receiving an instruction of playing the media file. Because an optimization downloading method of dividing the media file and a play method of playing the fragments are used, the downloading method is optimized. Moreover, the media file may also be played according to an order of the fragments. Even if when other fragments of the media file are not downloaded completely, the downloaded fragments may also be normally played. In this way, a user may play a part of the media file that is downloaded completely at any time, which achieves an objective of simply and quickly playing a media file that is not downloaded completely.

[0031] FIG. **2** is a flowchart of steps of a method for downloading and playing a media file according to another embodiment of the present application. FIG. **2** is an improvement on the basis of FIG. **1**. Referring to FIG. **2**, the method for downloading and playing a media file in this embodiment of the present application may include the following steps S**201**-S**225**:

[0032] Step S**201**: A client initiates a request for acquiring a media file to a server.

[0033] In this step, the media file may be a file such as music or a video. The client may be a mobile client such as a mobile phone.

[0034] Step S**203**: The server determines whether a file size of the media file is greater than a preset value, and executes step S**205** if the file size of the media file is greater than the preset value, or uses the media file as one fragment and executes step S**209** if the file size of the media file is not greater than the preset value.

[0035] In this step, the server may calculate a file size of the media file and determine whether the file size of the media file is greater than the preset value (for example, the preset value is a value such as 120 MB). The preset value may be set according to actual needs.

[0036] Step S**205**: The server divides the media file into an ordered list of fragments according to the current network connection information of the client, stores the fragments so as to form download list information of the fragments, and further sends the download list information of the fragments to the client.

[0037] In this step, a method of dividing the media file into the several fragments may be a method of performing average division according to a file size of the media file, or any other dividing method. If the average dividing method is used, the method may be that: if a media file that is 128 MB is divided into two fragments, the file size of each fragment is 64 MB.

[0038] The download list information of each of the fragments includes information such as a URL stored in the fragment.

[0039] Step S**207**: The client receives and stores the download list information of the fragments, and initiates a request for acquiring fragments of the media file to the server according to an order of the fragments in the download list information received from the server.

[0040] In this step, the order of the fragments may be a chronological order of the fragments (i.e., a chronological order in which the media file is played). For example, a fragment in which a start location of the media file is located is the first fragment; the fragments are sequentially arranged according to a play chronological order; and other fragments are sequentially the second fragment, the third fragment, and the like.

[0041] Step S**209**: The server returns the fragments of the media file to the client sequentially according to the request for acquiring the fragments of the media file received from the client.

[0042] If the file size of the media file acquired in S**203** is less than the preset value, in this step, the server uses the media file as one fragment and returns the fragment (i.e., the media file) to the client, that is, the server returns the media file to the client.

[0043] Step S**211**: The client determines whether a last play record of the media file is stored after acquiring the fragments of the media file returned by the server and receiving an instruction of playing the media file, and executes step S**213** if the last play record of the media file is stored, or executes step S**215** if the last play record of the media file is not stored.

[0044] In this step, a memory of the client prestores the last play record, and the last play record may record playback location information such as a last play time, a last play progress, and till which fragment a media file is played for the last time. For example, the instruction of playing the media file is sent when the client opens the media file and chooses to play the media file.

[0045] Step S**213**: Determine to play the media file from a location recorded by the last play record of the media file, use the location recorded by the last play record as an initial playback location, identify a fragment including the location recorded by the last play record, and execute step S**217**.

[0046] Step S**215**: Determine to play the media file from the start location of the media file, use the start location of the media file as the initial playback location, find the fragment including the start location of the media file is located, and execute step S**217**.

[0047] In this step, the fragment including the start location of the media file is located is the first fragment.

[0048] Step S**217**: Read the identified fragment, determine whether the identified fragment is complete, and execute step S**219** if the identified fragment is complete, or execute step S**221** if the identified fragment is incomplete.

[0049] In this step, whether the identified fragment is complete may be determined by determining whether the identified fragment includes a terminator. Specifically, it is determined that the identified fragment is complete if the identified fragment includes a terminator, and it is determined that the identified fragment is incomplete if the identified fragment does not include a terminator.

[0050] Step S**219**: Start to play the identified fragment from the initial playback location and execute step S**222**.

[0051] In this step, the identified fragment is played from a playback location recorded by the last play record of the fragment if the initial playback location is the location recorded by the last play record. The read fragment is played from the start location of the media file if the initial playback location is the start location of the media file.

[0052] Step S**221**: Skip check information of the identified fragment and only read media content of the identified fragment when reading the identified fragment, play the media content of the identified fragment from the initial playback location, and stop playing till there is no media information.

[0053] In this step, the check information may be stored in a file header of the fragment. Check information of the file header may include information such as time information of the fragment and a file size of the fragment. For example, the time information of the fragment in the check information may be displayed on a playback interface to be viewed by a user. If the fragment is incomplete, the check information of the file header of the fragment is generally also incomplete. The media information of the fragment is media data that may be played. An initial playback location of the media content of the identified fragment is the initial playback location of the media file. That is, the identified fragment is played from a playback location recorded by the last play record of the fragment if the initial playback location is the location recorded by the last play record. The read fragment is played from the start location of the media file if the initial playback location is the start location of the media file.

[0054] Step S**222**: Find a next fragment before playback of a currently played fragment is finished, determine whether the next fragment is complete, and execute step S**223** if the next fragment is complete, or execute step S**225** if the next fragment is incomplete.

[0055] In this step, the currently played fragment is a fragment that is being played. In a period of time (for example, 10 minutes) before the playback of the currently played fragment is finished, the next fragment is found and read, and it is determined whether the next fragment is complete. How to play the fragment in conditions in which the fragment is complete and the fragment is incomplete is set, so as to get ready for preparatory work of playing the next fragment in advance, which is mainly used for ensuring fluency of the playback.

[0056] Step S**223**: Play the next fragment when the playback of the currently played fragment is finished, and execute step S**222**.

[0057] Step S**225**: Skip check information of the next fragment and only read media content of the next fragment when reading the next fragment, play the media content of the next fragment when the playback of the currently played fragment is finished, and stop playing till there is no media information.

[0058] In this step, the check information may be stored in a file header of the fragment.

[0059] In this embodiment of the present application and in the present disclosure, the server divides the media file into an ordered list of fragments according to the current network connection information of the client and stores the fragments so as to form the download list information of the fragments; and the client initiates a request for acquiring fragments of the media file to the server according to an order of the fragments in download list information, and plays the fragments of the media file according to the order of the fragments when receiving an instruction of playing the media file. Moreover, whether a last play record of the media file is stored in a memory is determined. The location recorded by the last play record is used as an initial playback location, and a fragment including the location recorded by the last play record in the download list information is found if the last play record of the media file is stored; or the start location of the media file is used as the initial playback location, and the fragment including the start location of the media file is located is found if the last play record of the media file is not stored. The identified fragment is read, and whether the identified fragment is complete is determined. The read fragment starts to be played from the initial playback location if the identified fragment is complete. The media content of the identified fragment is played from the initial playback location if the identified fragment is incomplete. Because an optimization downloading method of dividing the media file and a play method of playing the downloaded complete and incomplete fragments are used, the downloading method is optimized. Moreover, the media file may also be played according to an order of the fragments. Even if when other fragments of the media file are not downloaded completely, the downloaded complete and downloaded incomplete fragments may also be normally played. In this way, a user may play a part of the media file that is downloaded completely at any time, which achieves an objective of simply and quickly playing a media file that is not downloaded completely. Moreover, after playback of a currently played fragment is finished, a next fragment can further be fluently played, and unstable phenomena such as a stop and intermission during the playback do not arise, thereby ensuring excellent feelings of the user during entertainment, improving a playback quality of the media file, and improving user playback experiences of the media file.

[0060] FIG. **3** is a block diagram of main architecture of a system for downloading and playing a media file according to an embodiment of the present application.

[0061] Referring to FIG. **3**, this embodiment of the present application provides a client **311**. The client **311** includes one or more processors, memory and a plurality of program modules stored in the memory and to be executed by the one or more processors, the plurality of program modules further including:

[0062] a file acquiring module **301**, configured to initiate a request for acquiring a media file to a server, the request including current network connection information of the client, and receive download list information from the server, wherein the download list information identifies an ordered list of fragments extracted from the media file by the server in accordance with the current network connection information of the client;

[0063] a fragment acquiring module **305**, configured to initiate a request for acquiring the fragments of the media file to the server according to the order of the fragments in the download list information, and acquire the fragments of the media file returned by the server; and

[0064] a playback module **309**, configured to play the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

[0065] The playback module **309** is further configured to:

[0066] determine whether a last play record of the media file is stored;

[0067] determine that a location recorded by the last play record is an initial playback location and identify a fragment including the location recorded by the last play record in the download list information if the last play record of the media file is stored;

[0068] determine that a start location of the media file is the initial playback location and identify a fragment including the start location of the media file in the download list information if the last play record of the media file is not stored; and

[0069] determine whether the identified fragment is complete; and start to play the identified fragment from the initial playback location if it is determined that the identified fragment is complete.

[0070] The playback module **309** is further configured to skip check information of the identified fragment, only read media content of the identified fragment when reading the identified fragment, and play the media content of the identified fragment from the initial playback location if it is determined that the identified fragment is incomplete after determining whether the identified fragment is complete.

[0071] The playback module **309** is further configured to: after receiving an instruction of playing the media file, identify a next fragment before playback of a currently played fragment is finished, and determine whether the next fragment is complete;

[0072] play the next fragment when the playback of the currently played fragment is finished if it is determined that the next fragment is complete; or

[0073] skip check information of the next fragment, only read media content of the next fragment when reading the next fragment, and play the media content of the next fragment when the playback of the currently played fragment is finished if it is determined that the next fragment is incomplete.

[0074] This embodiment of the present application further provides a server **321**. The server **321** includes one or more processors, memory and a plurality of program modules stored in the memory and to be executed by the one or more processors, the plurality of program modules further including:

[0075] a fragmentation module **303**, configured to divide a media file into an ordered list of fragments according to the current network connection information of the client after receiving a request for acquiring the media file from a client, store the fragments so as to form download list information of the fragments, and further send the download list information of the fragments to the client; and

[0076] a fragment returning module **307**, configured to return the fragments of the media file to the client sequentially according to a request for acquiring the fragments of the media file received from the client.

[0077] The fragmentation module **303** is further configured to:

[0078] determine whether file size of the media file is greater than a preset value;

[0079] divide the media file into the several fragments if the file size of the media file is greater than the preset value; or

[0080] use the media file as one fragment and return the fragment to the client if the file size of the media file is not greater than the preset value.

[0081] Referring to FIG. **3**, the system for downloading and playing a media file includes a client **311** and a server **321**. The client **311** further includes a file acquiring module **301**, a fragment acquiring module **305**, and a playback module **309**. The server **321** includes a fragmentation module **303** and a fragment returning module **307**.

[0082] Specifically, the file acquiring module **301** that may be set in the client is configured to initiate a request for acquiring a media file to a server, the request including current network connection information of the client. The media file may be a file such as music or a video.

[0083] The fragment dividing module **303** that may be set in the server is configured to divide the media file into an ordered list of fragments according to the current network connection information of the client after receiving the request for acquiring the media file from the client, store the fragments so as to form download list information of the fragments, and further send the download list information of the fragments to the client.

[0084] Moreover, the fragmentation module **303** is further configured to determine whether file size of the media file is greater than a preset value; divide the media file into the several fragments if the file size of the media file is greater than the preset value; or use the media file as one fragment and return the fragment to the client if the file size of the media file is not greater than the preset value.

[0085] The segment acquiring module **305** that may be set in the client is configured to initiate a request for acquiring the fragments of the media file to the server according to an order of the fragments in download list information received from the server, and acquire the fragments of the media file returned by the server.

[0086] The fragment returning module **307** that may be set in the server is configured to return the fragments of the media file to the client sequentially according to the request for acquiring the fragments of the media file received from the client.

[0087] The playback module **309** that may be set in the client is configured to play the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

[0088] Moreover, the playback module **309** is further configured to determine whether a last play record of the media file is stored in a memory; determine to play the media file from a location recorded by the last play record of the media file, use the location recorded by the last play record as an initial playback location, and identify a fragment including the location recorded by the last play record in the download list information if the last play record of the media file is stored; or determine to play the media file from the start location of the media file, use the start location of the media file as the initial playback location, and find the fragment including the start location of the media file is located if the last play record of the media file is not stored; read the identified fragment, and determine whether the identified fragment is complete; and start to play the identified fragment from the initial playback location if the identified fragment is complete.

[0089] Moreover, the playback module **309** is further configured to skip check information of the identified fragment and only read media content of the identified fragment when reading the identified fragment, and play the media content of the identified fragment from the initial playback location if it is determined that the identified fragment is incomplete.

[0090] Moreover, the playback module **309** is further configured to identify a next fragment before playback of a currently played fragment is finished, and determine whether the next fragment is complete; play the next fragment when the playback of the currently played fragment is finished if the next fragment is complete; or skip check information of the next fragment, only read media content of the next fragment when reading the next fragment, and play the media content of the next fragment when the playback of the currently played fragment is finished if the next fragment is incomplete.

[0091] In this embodiment of the present application and in the present disclosure, the server divides the media file into an ordered list of fragments according to the current network connection information of the client and stores the fragments so as to form the download list information of the fragments; and the client initiates a request for acquiring fragments of the media file to the server according to an order of the fragments in download list information, and plays the fragments of the media file according to the order of the fragments when receiving an instruction of playing the media file. Moreover, whether a last play record of the media file is stored in a memory is determined. The location recorded by the last play record is used as an initial playback location, and a fragment including the location recorded by the last play record in the download list information is found if the last play record of the media file is stored; or the start location of the media file is used as the initial playback location, and the fragment including the start location of the media file is located is found if the last play record of the media file is not stored. The identified fragment is read, and whether the identified fragment is complete is determined. The read fragment starts to be played from the initial playback location if the identified fragment is complete. The media content of the identified fragment is played from the initial playback location if the identified fragment is incomplete. Because an optimization downloading method of dividing the media file and a play method of playing the downloaded complete and incomplete fragments are used, the downloading method is optimized. Moreover, the media file may also be played according to an order of the fragments. Even if when other fragments of the media file are not downloaded completely, the downloaded complete and downloaded incomplete fragments may also be normally played. In this way, a user may play a part of the media file that is downloaded completely at any time, which achieves an objective of simply and quickly playing a media file that is not downloaded completely. Moreover, after playback of a currently played fragment is finished, a next fragment can further be fluently played, and unstable phenomena such as a stop and intermission during the playback do not arise, thereby ensuring excellent feelings of users during entertainment, improving a playback quality of the media file, and improving user playback experiences of the media file.

[0092] A person of ordinary skill in the art may understand that all or some of the processes of the methods in the foregoing embodiments may be implemented by a computer program controlling relevant hardware. The program may be stored in a computer readable storage medium. During the execution of the program, processes of the foregoing method embodiments may be included. The storage medium may be a magnetic disk, an optical disc, a read-only memory (ROM), a random access memory (RAM) or the like.

[0093] Therefore, the embodiments of the present application further provide one or more storage media containing a computer executable instruction, the computer executable instruction being used for executing a method running on a client for downloading and playing a media file, and the method including the following steps:

[0094] initiating a request for acquiring a media file to a server, the request including current network connection information of the client;

[0095] receiving download list information from the server, wherein the download list information identifies an ordered list of fragments extracted from the media file by the server in accordance with the current network connection information of the client;

[0096] initiating a request for acquiring the fragments of the media file to the server according to the order of the fragments in the download list information, and acquiring the fragments of the media file returned by the server; and

[0097] playing the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

[0098] The embodiments of the present application further provide another one or more storage media containing a computer executable instruction, the computer executable instruction being used for executing a method running on a server for downloading and playing a media file, and the method including the following steps:

[0099] dividing the media file into an ordered list of fragments according to the current network connection information of the client after receiving a request for acquiring the media file from a client, storing the fragments so as to form download list information of the fragments, and further sending the download list information of the fragments to the client; and

[0100] returning the fragments of the media file to the client sequentially according to a request for acquiring the fragments of the media file received from the client.

[0101] The above descriptions are merely preferred embodiments of the present application, and are not intended to limit the present application in any form. Although the present application has been disclosed above through the preferred embodiments, the embodiments are not intended to limit the present application. A person skilled in the art can make some equivalent variations, alterations or modifications to the above-disclosed technical content without departing from the scope of the technical solutions of the present application to obtain equivalent embodiments. Any simple alteration, equivalent change or modification made to the above embodiments according to the technical essence of the present application without departing from the content of the technical solutions of the present application shall fall within the scope of the technical solutions of the present application.

What is claimed is:

1. A method for downloading and playing a media file, comprising:

at a client having one or more processors and memory storing program modules to be executed by the one or more processors:

initiating a request for acquiring a media file to a server, the request including current network connection information of the client;

receiving download list information from the server, wherein the download list information identifies an ordered list of fragments extracted from the media file by the server in accordance with the current network connection information of the client;

initiating a request for acquiring the fragments of the media file to the server according to the order of the fragments in the download list information, and acquiring the fragments of the media file returned by the server; and

playing the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

2. The method according to claim **1**, further comprising:

after receiving an instruction of playing the media file:

determining whether a last play record of the media file is stored;

determining to use a location recorded by the last play record as an initial playback location and identifying a fragment including the location recorded by the last play record in the download list information if the last play record of the media file is stored;

determining to use a start location of the media file as the initial playback location and identifying a fragment including the start location of the media file in the download list information if the last play record of the media file is not stored;

determining whether the identified fragment is complete; and

starting to play the identified fragment from the initial playback location if it is determined that the identified fragment is complete.

3. The method according to claim **2**, further comprising:

after determining whether the identified fragment is complete:

skipping check information of the identified fragment and only reading media content of the identified fragment when reading the identified fragment, and playing the media content of the identified fragment from the initial playback location if it is determined that the identified fragment is incomplete.

4. The method according to claim **2**, comprising:

after receiving an instruction of playing the media file:

identifying a next fragment before playback of a currently played fragment is finished and determining whether the next fragment is complete;

playing the next fragment when the playback of the currently played fragment is finished if it is determined that the next fragment is complete; or

skipping check information of the next fragment and only reading media content of the next fragment when reading the next fragment, and playing the media content of the next fragment when the playback of the currently played fragment is finished if it is determined that the next fragment is incomplete.

5. The method according to claim **4**, wherein the next fragment is immediately after the currently played fragment.

6. The method according to claim **1**, wherein sizes of the fragments of the media file are dynamically determined by the current network connection information of the client.

7. A method for downloading a media file, comprising:

at a server having one or more processors and memory storing program modules to be executed by the one or more processors:

receiving a request for acquiring a media file from a client, the request including current network connection information of the client;

dividing the media file into an ordered list of fragments according to the current network connection information of the client, storing the fragments so as to form download list information of the fragments, and further sending the download list information of the fragments to the client;

receiving a request for acquiring the fragments of the media file from the client; and

returning the fragments of the media file to the client sequentially according to the order of the fragments in the download list information.

8. The method according to claim **7**, wherein the division of the media file into an ordered list of fragments according to the current network connection information of the client further comprises the following steps:

determining whether a file size of the media file is greater than a preset value;

dividing the media file into the several fragments if the file size of the media file is greater than the preset value; or

using the media file as one fragment and returning the fragment to the client if the file size of the media file is not greater than the preset value.

9. A client, comprising:

one or more processors;

memory; and

a plurality of program modules, wherein the plurality of program modules, when executed by the one or more processors, cause the client to perform predefined operations, the plurality of program modules further including:

a file acquiring module, configured to initiate a request for acquiring a media file to a server, the request including current network connection information of the client, and receive download list information from the server, wherein the download list information identifies an ordered list of fragments extracted from the media file by the server in accordance with the current network connection information of the client;

a fragment acquiring module, configured to initiate a request for acquiring the fragments of the media file to the server according to the order of the fragments in the download list information, and acquire the fragments of the media file returned by the server; and

a playback module, configured to play the fragments of the media file returned by the server in accordance with the order of the fragments according to a received instruction of playing the media file.

10. The client according to claim **9**, wherein the playback module is further configured to:

determine whether a last play record of the media file is stored;

determine that a location recorded by the last play record is an initial playback location and identify a fragment including the location recorded by the last play record in the download list information if the last play record of the media file is stored;

determine that a start location of the media file is the initial playback location and identify a fragment including the start location of the media file in the download list information if the last play record of the media file is not stored;

determine whether the identified fragment is complete; and

start to play the identified fragment from the initial playback location if it is determined that the identified fragment is complete.

11. The client according to claim **9**, wherein the playback module is further configured to:

skip check information of the identified fragment and only read media content of the identified fragment when reading the identified fragment, and play the media content of the identified fragment from the initial playback location if it is determined that the identified fragment is incomplete after determining whether the identified fragment is complete.

12. The client according to claim **9**, wherein the playback module is further configured to:

after receiving an instruction of playing the media file, identify a next fragment before playback of a currently played fragment is finished, and determine whether the next fragment is complete;

play the next fragment when the playback of the currently played fragment is finished if it is determined that the next fragment is complete; or

skip check information of the next fragment and only read media content of the next fragment when reading the next fragment, and play the media content of the next fragment when the playback of the currently played fragment is finished if it is determined that the next fragment is incomplete.

13. The client according to claim **12**, wherein the next fragment is immediately after the currently played fragment.

14. The client according to claim **9**, wherein sizes of the fragments of the media file are dynamically determined by the current network connection information of the client.

* * * * *