Abstract: A data processing method includes: determining, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response; sending a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and displaying the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response; and receiving the second-type action data sent by the server, and displaying the second-type action data.
DATA PROCESSING METHOD, TERMINAL AND SERVER

FIELD OF THE TECHNOLOGY

[0001] The present disclosure relates to the field of data processing technologies, and in particular, to a data processing method, terminal and server.

BACKGROUND OF THE DISCLOSURE

[0002] With the rapid development of the Internet, online games are more and more popular among players; action games are dominating online games, and emphasize real-time interaction and smooth experience, and a processing latency should be shorter than a physical response time of a user. The processing latency is a time from the moment when the user initiates an action instruction to the moment when a system responds correspondingly. All traditional action games are of standalone versions, in which all computations and interactions are completed locally, and as long as a central processing unit (CPU) and memory meet requirements of software, it can be ensured that the processing latency is shorter than the response time of the user. However, online action games need network connections to implement synchronous interactions of different players, and inherent properties of a network, namely, latency and fluctuation, directly affect the value of the response latency in the game. Online game players are dispersed all over the country; network access conditions are complicated, and latencies and fluctuations vary significantly, making latency processing more complicated.

[0003] A long processing latency severely affects the experience of synchronous interaction in the game, causing a feeling of stickiness to the user; the feeling of stickiness is used to describe a situation in which the processing latency of a game system consisting of a terminal and a server exceeds a response time of a player, and the player feels that the game responds slowly. Consequently, communication efficiency between the user and a game application is lowered.

SUMMARY

[0004] A first aspect of the present disclosure provides a data processing method, including: determining, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response;
sending a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and displaying the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response; and

receiving the second-type action data sent by the server, and displaying the second-type action data.

[0005] A second aspect of the present disclosure provides a data processing method, including:

receiving a data requesting message sent by a terminal when the terminal displays first-type action data, the data requesting message being used for requesting second-type action data corresponding to a specified skill specified by a user, the first-type action data being data that can be displayed before a server makes a response, and the second-type action data being data that cannot be displayed until the server makes a response;

determining, according to the data requesting message, the second-type action data corresponding to the specified skill; and

sending the second-type action data to the terminal.

[0006] A third aspect of the present disclosure provides a terminal, including:

a receiving unit, configured to receive a casting instruction input by a user for a specified skill;

a determining unit, configured to determine first-type action data corresponding to the specified skill received by the receiving unit, the first-type action data being data that can be displayed before a server makes a response;

a sending unit, configured to send a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill received by the receiving unit; and

a display unit, configured to display the first-type action data determined by the determining unit, the second-type action data being data that cannot be displayed until the server makes a response;

the receiving unit being further configured to receive the second-type action data sent by the server; and
the display unit being further configured to display the second-type action data received by the receiving unit.

[0007] A fourth aspect of the present disclosure provides a server, including:

1. a receiving unit, configured to receive a data requesting message sent by a terminal when the terminal displays first-type action data, the data requesting message being used for requesting second-type action data corresponding to a specified skill specified by a user, the first-type action data being data that can be displayed before the server makes a response, and the second-type action data being data that cannot be displayed until the server makes a response;
2. a determining unit, configured to determine, according to the data requesting message received by the receiving unit, the second-type action data corresponding to the specified skill; and
3. a sending unit, configured to send, to the terminal, the second-type action data determined by the determining unit.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] To illustrate the technical solutions in the embodiments of the present invention more clearly, the following briefly introduces the accompanying drawings required for describing the embodiments of the present invention. Apparently, the accompanying drawings in the following description show merely some embodiments of the present invention, and a person skilled in the art may still derive other drawings from these accompanying drawings without creative efforts.

[0009] FIG. 1 is a schematic diagram of an embodiment data processing method according to the embodiments of the present invention;

[0010] FIG. 2 is a schematic diagram of another embodiment of a data processing method according to the embodiments of the present invention;

[0011] FIG. 3 is a schematic diagram of another embodiment of a data processing method according to the embodiments of the present invention;

[0012] FIG. 4 is a schematic diagram of an embodiment of a terminal according to the embodiments of the present invention;

[0013] FIG. 5 is a schematic diagram of another embodiment of a terminal according to the embodiments of the present invention;
FIG. 6 is a schematic diagram of another embodiment of a terminal according to the embodiments of the present invention;

FIG. 7 is a schematic diagram of an embodiment of a server according to the embodiments of the present invention;

FIG. 8 is a schematic diagram of another embodiment of a server according to the embodiments of the present invention;

FIG. 9 is a schematic diagram of another embodiment of a terminal according to the embodiments of the present invention;

FIG. 10 is a schematic diagram of another embodiment of a server according to the embodiments of the present invention; and

FIG. 11 is a schematic diagram of an embodiment of a data processing system according to the embodiments of the present invention.

DESCRIPTION OF EMBODIMENTS

An embodiment of the present invention provides a data processing method, which can avoid the feeling of stickiness during a game process in a case where a network latency cannot be changed, thereby improving communication efficiency between a user and a game application. Embodiments of the present invention further provide a corresponding terminal, server, and system. The following separately describes the data processing method, terminal, server, and system in detail.

The technical solutions in the embodiments of the present application are described clearly and completely below with reference to the accompanying drawings in the embodiments of the present application. Apparently, the embodiments described are merely some embodiments rather than all embodiments of the present invention. All other embodiments obtained by a person skilled in the art based on the embodiments in the present invention without creative efforts belong to the protection scope of the present disclosure.

Referring to FIG. 1, an embodiment of a data processing method provided by the embodiments of the present invention includes:

Determine, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response.
A player usually plays a role of a virtual character in an online game, and the character has corresponding skills such as boxing, kicking, slashing, and the player may cast different skills by pressing different keys on a keyboard; each skill generally may be divided into three parts, namely, Chant, OnCast, and OnHit, and all types of skills begin with the phase of "Chant" or "OnCast"; the phases of "Chant" and "OnCast" purely present properties, and can be executed without waiting for data replied by the server.

The first-type action data in this embodiment of the present invention is equivalent to "Chant" or "OnCast" data, for example: in one scenario, the virtual character of the player in the game needs to draw a sword and swing the sword downward to slash the enemy, the process of drawing a sword is equivalent to the first-type action data, namely, the "Chant" or "OnCast" data.

102: Send a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and display the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response.

In this embodiment of the present invention, the second-type action data is action data with a substantial manifestation, and is equivalent to the part of "OnHit" in skill division. For example, in one game scenario, the virtual character of the player in the game needs to draw a sword and swing the sword downward to slash the enemy, where the action of swinging the sword downward to slash the enemy is the second-type action data.

The terminal starts displaying the first-type action data when requesting the second-type action data; in this way, the user almost does not feel any waiting time in terms of visual experience, and the terminal starts displaying data immediately after the user delivers the instruction, which reduces, in another form, a network latency.

103: Receive the second-type action data sent by the server, and display the second-type action data.

In this embodiment of the present invention, after a casting instruction input by a user for a specified skill is received, first-type action data corresponding to the specified skill is determined, the first-type action data being data that can be displayed before a server makes a response; a data requesting message is sent to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and the first-type action data is displayed, the second-type action data being data that cannot be displayed until the server makes a response; and the second-type action data sent by the server is received, and the second-type
action data is displayed. Compared with the existing technology, when requesting the second-type action data from the server, the data processing method provided by this embodiment of the present invention may first display the first-type action data that can be displayed before the server makes a response, to avoid the feeling of stickiness during a game process, thereby improving communication efficiency between a user and a game application.

[0031] Optionally, on the basis of the embodiment corresponding to FIG. 1, in another embodiment of the data processing method provided by the embodiments of the present invention, when the first-type action data is displayed, the method may further include:

listening, to determine whether the second-type action data is received.

[0032] Optionally, on the basis of the optional embodiment corresponding to FIG. 1, in another embodiment of the data processing method provided by the embodiments of the present invention,

the terminal buffers the second-type action data if it is detected by means of listening that the second-type action data has arrived before display of the first-type action data is finished.

[0033] Optionally, on the basis of the optional embodiment corresponding to FIG. 1, in another embodiment of the data processing method provided by the embodiments of the present invention,

the terminal continues to display preset redundant action data associated with the first-type action data if it is detected by means of listening that the second-type action data has not arrived yet when display of the first-type action data is finished.

[0034] In the implementation of the present disclosure, by using a game scenario as an example, when the virtual character of the player in the game needs to draw a sword and swing the sword downward to slash the enemy, the terminal listens, when displaying data about the process of drawing a sword, to detect whether data about swinging the sword downward to slash the enemy arrives.

[0035] When it is detected by means of listening that the data about swinging the sword downward to slash the enemy has arrived, the terminal first buffers the data about swinging the sword downward to slash the enemy, and after finishing displaying the data about drawing a sword, the terminal directly displays the buffered data about swinging the sword downward to slash the enemy.
If the data about swinging the sword downward to slash the enemy has not arrived yet by the time display of the data about drawing a sword is finished, the terminal further displays the preset redundant action data associated with the data about drawing a sword, where the redundant action data may be picking up the sword with both hands and putting it in front, and if the data about swinging the sword downward to slash the enemy arrives at this time, the terminal may continue to display the data about swinging the sword downward to slash the enemy. In this way, the player does not feel any pause in the entire process in terms of visual experience, which reduces the feeling of freeze in the game process, where the feeling of freeze is used to describe a situation where the processing latency of a processing system consisting of a terminal and a server is unstable, and the processing is sometimes prompt and sometimes slow, which severely affects user experience.

In this way, the player neither has the feeling of stickiness nor the feeling of freeze during the game process; the game is made smoother, and the experience of the player when playing the game is also enhanced.

Referring to FIG. 2, another embodiment of a data processing method provided by the embodiments of the present invention includes:

Receive a data requesting message sent by a terminal when the terminal displays first-type action data, the data requesting message being used for requesting second-type action data corresponding to a specified skill specified by a user, the first-type action data being data that can be displayed before a server makes a response, and the second-type action data being data that cannot be displayed until the server makes a response.

A player usually plays a role of a virtual character in an online game, and the character has corresponding skills such as boxing, kicking, slashing, and the player may cast different skills by pressing different keys on a keyboard; each skill generally may be divided into three parts, namely, Chant, OnCast, and OnHit, and all types of skills begin with the phase of "Chant" or "OnCast"; the phases of "Chant" and "OnCast" purely present properties, and can be executed without waiting for data replied by the server.

The first-type action data in this embodiment of the present invention is equivalent to "Chant" or "OnCast" data, for example: in one scenario, the virtual character of the player in the game needs to draw a sword and swing the sword downward to slash the enemy, the process of drawing a sword is equivalent to the first-type action data, namely, the "Chant" or "OnCast" data.
The second-type action data is action data with a substantial manifestation, and is equivalent to the part of "OnHit" in skill division. For example, in one game scenario, the virtual character of the player in the game needs to draw a sword and swing the sword downward to slash the enemy, where the action of swinging the sword downward to slash the enemy is the second-type action data.

The terminal starts displaying the first-type action data when requesting the second-type action data; in this way, the user almost does not feel any waiting time in terms of visual experience, and the terminal starts displaying data immediately after the user delivers the instruction, which reduces, in another form, a network latency.

202: Determine, according to the data requesting message, the second-type action data corresponding to the specified skill.

203: Send the second-type action data to the terminal.

In this embodiment of the present invention, a data requesting message sent by a terminal is received, the data requesting message being used for requesting second-type action data corresponding to a specified skill specified by a user, and the second-type action data being data that cannot be displayed until a server responds; the second-type action data corresponding to the specified skill is determined according to the data requesting message; and the second-type action data is sent to the terminal. As compared with the existing technology, in the data processing method provided by this embodiment of the present invention, when requesting the second-type action data from the server, the terminal may first display the first-type action data that can be displayed before the server makes a response, to avoid the feeling of stickiness during a game process, thereby improving communication efficiency between a user and a game application.

Optionally, on the basis of the embodiment corresponding to FIG. 2, in another embodiment of the data processing method provided by the embodiments of the present invention, after the receiving a data request message sent by a terminal, the method may further include:

buffering the data requesting message and recording an expiration time of the data requesting message when determining that the data requesting message arrives at the server ahead of time; and

reading the data requesting message when the expiration time arrives.

In this embodiment of the present application, if presentation of second-type action data corresponding to a previous is not finished yet when the data requesting message arrives at the
server ahead of time, the server buffers the current data requesting message and records the expiration time of the data requesting message. For example, a timer may be set, and when the timer expires, the server reads the data requesting message, determines the second-type action data corresponding to the specified skill according to the data requesting message, and sends the second-type action data to the terminal. The server buffers the request that arrives ahead of time, to avoid that the data requesting message arriving earlier than expected is rejected.

[0049] Before the presentation of the second-type action data corresponding to the previous request is finished, the server does not process the next request, because the server needs to control the frequency and rhythm for casting skills, to effectively prevent the player from cheating.

[0050] Referring to FIG. 3, another embodiment of a data processing method provided by the embodiments of the present invention includes:

[0051] S300: A terminal receives a casting instruction input by a user for a specified skill.

[0052] S305: The terminal determines first-type action data corresponding to the specified skill.

[0053] The first-type action data is data that can be displayed before a server makes a response.

[0054] S310: The terminal sends a data request message to a server.

[0055] The data requesting message is used for requesting second-type action data corresponding to the specified skill, and the second-type action data is data that cannot be displayed until the server makes a response.

[0056] S315: The terminal displays the first-type action data.

[0057] The sequence of S310 and S315 is not limited.

[0058] S320: The server buffers the data requesting message when determining that the data requesting message arrives ahead of time.

[0059] 325: Determine the second-type action data corresponding to the specified skill according to the data requesting message when an expiration time of the data requesting message arrives.

[0060] S330: The server sends the second-type action data to the terminal.

[0061] S335: The terminal buffers the second-type action data when determining that the second-type action data arrives ahead of time.
S340: Continue to display preset redundant action data associated with the first-type action data when the second-type action data arrives later than expected.

S345: Display the second-type action data.

Compared with the existing technology, when requesting the second-type action data from the server, the data processing method provided by this embodiment of the present invention may first display the first-type action data that can be displayed before the server makes a response, to avoid the feeling of stickiness during a game process, thereby improving communication efficiency between a user and a game application. Moreover, a bidirectional buffer technology and a data redundancy technology are used on the terminal and the server, thereby also resolving the feeling of freeze caused by the network instability.

Referring to FIG. 4, an embodiment of a terminal 40 provided by the embodiments of the present invention includes:

- A receiving unit 401, configured to receive a casting instruction input by a user for a specified skill;
- A determining unit 402, configured to determine first-type action data corresponding to the specified skill received by the receiving unit 401, the first-type action data being data that can be displayed before a server makes a response;
- A sending unit 403, configured to send a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill received by the receiving unit 401; and
- A display unit 404, configured to display the first-type action data determined by the determining unit 402, the second-type action data being data that cannot be displayed until the server makes a response;
- The receiving unit 401 being further configured to receive the second-type action data sent by the server; and
- The display unit 404 being further configured to display the second-type action data received by the receiving unit 401.

In this embodiment, the receiving unit 401 receives a casting instruction input by a user for a specified skill; the determining unit 402 determines first-type action data corresponding to the specified skill received by the receiving unit 401, the first-type action data being data that can be displayed before a server makes a response; the sending unit 403 sends a data requesting message to
the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill received by the receiving unit 401; the display unit 404 displays the first-type action data determined by the determining unit 402, the second-type action data being data that cannot be displayed until the server makes a response; the receiving unit 401 further receives the second-type action data sent by the server; and the display unit 404 is further configured to display the second-type action data received by the receiving unit 401. Compared with the existing technology, when requesting the second-type action data from the server, the terminal provided by this embodiment of the present invention may first display the first-type action data that can be displayed before the server makes a response, to avoid the feeling of stickiness during a game process, thereby improving communication efficiency between a user and a game application.

[0067] Optionally, on the basis of the embodiment corresponding to FIG. 4, referring to FIG. 5, in another embodiment of the terminal provided by the embodiments of the present invention, the terminal 40 further includes:

a listening unit 405, configured to listen, when the display unit 404 displays the first-type action data, to determine whether the second-type action data is received.

[0068] Optionally, on the basis of the embodiment corresponding to FIG. 5, referring to in FIG. 6, in another embodiment of the terminal provided by the embodiments of the present invention, the terminal 40 further includes:

a buffering unit 406, configured to buffer the second-type action data if the listening unit 405 detects by means of listening that the second-type action data has arrived before display of the first-type action data is finished.

[0069] Optionally, on the basis of the optional embodiment corresponding to FIG. 5, in another embodiment of the terminal provided by the embodiments of the present invention, the display unit is further configured to continue to display preset redundant action data associated with the first-type action data if the listening unit 405 detects by means of listening that the second-type action data has not arrived yet when display of the first-type action data is finished.

[0070] Referring to FIG. 7, an embodiment of a server 50 provided by the embodiments of the present invention includes:
a receiving unit 501, configured to receive a data requesting message sent by a
terminal when the terminal displays first-type action data, the data requesting message being used
for requesting second-type action data corresponding to a specified skill specified by a user, the
first-type action data being data that can be displayed before the server makes a response, and the
second-type action data being data that cannot be displayed until the server makes a response;

a determining unit 502, configured to determine, according to the data requesting
message received by the receiving unit 501, the second-type action data corresponding to the
specified skill; and

a sending unit 503, configured to send, to the terminal, the second-type action data
determined by the determining unit 502.

In this embodiment of the present invention, the receiving unit 501 receives a data
requesting message sent by a terminal when the terminal displays first-type action data, the data
requesting message being used for requesting second-type action data corresponding to a specified
skill specified by a user, the first-type action data being data that can be displayed before the server
makes a response, and the second-type action data being data that cannot be displayed until the
server makes a response; the determining unit 502 determines, according to the data requesting
message received by the receiving unit 501, the second-type action data corresponding to the
specified skill; and the sending unit 503 sends, to the terminal, the second-type action data
determined by the determining unit 502. As compared with the existing technology, with the server
provided by this embodiment of the present invention, when requesting the second-type action data
from the server, the terminal may first display the first-type action data that can be displayed before
the server makes a response, to avoid the feeling of stickiness during a game process, thereby
improving communication efficiency between a user and a game application.

Optionally, on the basis of the embodiment corresponding to FIG. 7, referring to in
FIG. 8, in another embodiment of the server provided by the embodiments of the present invention,
the server 50 further includes:

a buffering unit 504, configured to buffer the data requesting message when the
determining unit 502 determines that the data requesting message arrives at the server ahead of time;

a recording unit 505, configured to record an expiration time of the data requesting
message buffered by the buffering unit 504; and
a reading unit 506, configured to read the data requesting message buffered by the buffering unit 505 when the expiration time recorded by the recording unit 504 arrives.

[0073] Refer to FIG. 9, which shows a schematic structural diagram of a terminal involved in the embodiments of the present invention. The terminal may be configured to implement the data processing method provided in the foregoing embodiments. Specifically:

[0074] http://baike.baidu.com/view/1804790.htm The terminal 1200 may include components such as a radio frequency (RF) circuit 110, a memory 120 including one or more computer readable storage media, an input unit 130, a display unit 140, a sensor 150, an audio circuit 160, a Wireless Fidelity (WiFi) module 170, a processor 180 including one or more processing cores, and a power supply 190. A person skilled in the art may understand that, the structure of the terminal shown in FIG. 9 does not constitute a limitation to the terminal, and the terminal may include more components or fewer components than those shown in the figure, or some components may be combined, or a different component deployment may be used.

[0075] The RF circuit 110 may be configured to receive and send signals during an information sending and receiving process or a call process. Particularly, the RF circuit 1110 receives downlink information from a base station, then delivers the downlink information to the one or more processors 180 for processing, and sends related uplink data to the base station. Generally, the RF circuit 110 includes, but is not limited to, an antenna, at least one amplifier, a tuner, one or more oscillators, a subscriber identity module (SIM) card, a transceiver, a coupler, a low noise amplifier (LNA), and a duplexer. In addition, the RF circuit 110 may also communicate with a network and another device by wireless communication. The wireless communication may use any communications standard or protocol, which includes, but is not limited to, Global System for Mobile Communication (GSM), General Packet Radio Service (GPRS), Code Division Multiple Access (CDMA), Wideband Code Division Multiple Access (WCDMA), Long Term Evolution (LTE), e-mail, and Short Messaging Service (SMS).

[0076] The memory 120 may be configured to store a software program and module. The processor 180 runs the software program and module stored in the memory 120, to implement various functional applications and data processing. The memory 120 may mainly include a program storage area and a data storage area. The program storage area may store an operating system, an application program required by at least one function (such as a sound playback function and an image display function), and the like. The data storage area may store data (such as audio data and an address book) created according to use of the terminal 1200, and the like. In addition,
the memory 120 may include a high-speed random access memory, and may also include a non-volatile memory, such as at least one magnetic disk storage device, a flash memory, or another volatile solid-state storage device. Accordingly, the memory 120 may further include a memory controller, so that the processor 180 and the input unit 130 access the memory 120.

[0077] The input unit 130 may be configured to receive input digit or character information, and generate a keyboard, mouse, joystick, optical, or track ball signal input related to the user setting and function control. Specifically, the input unit 130 may include a touch-sensitive surface 131 and another input device 132. The touch-sensitive surface 131, which may also be referred to as a touch screen or a touch panel, may collect a touch operation of a user on or near the touch-sensitive surface (such as an operation of a user on or near the touch-sensitive surface 131 by using any suitable object or accessory, such as a finger or a stylus), and drive a corresponding connection apparatus according to a preset program. Optionally, the touch-sensitive surface 131 may include two parts: a touch detection apparatus and a touch controller. The touch detection apparatus detects a touch position of the user, detects a signal generated by the touch operation, and transfers the signal to the touch controller. The touch controller receives the touch information from the touch detection apparatus, converts the touch information into touch point coordinates, and sends the touch point coordinates to the processor 180. Moreover, the touch controller can receive and execute a command sent from the processor 180. In addition, the touch-sensitive surface 131 may be a resistive, capacitive, infrared, or surface acoustic wave type touch-sensitive surface. In addition to the touch-sensitive surface 131, the input unit 130 may further include another input device 132. Specifically, another input device 132 may include, but is not limited to, one or more of a physical keyboard, a functional key (such as a volume control key or a switch key), a track ball, a mouse, and a joystick.

[0078] The display unit 140 may be configured to display information input by the user or information provided for the user, and various graphical user interfaces of the terminal 1200. The graphical user interfaces may be formed by a graph, a text, an icon, a video, and any combination thereof. The display unit 140 may include a display panel 141. Optionally, the display panel 141 may be configured by using a liquid crystal display (LCD), an organic light-emitting diode (OLED), or the like. Further, the touch-sensitive surface 131 may cover the display panel 141. After detecting a touch operation on or near the touch-sensitive surface 131, the touch-sensitive surface 131 transfers the touch operation to the processor 180, so as to determine the type of the touch event. Then, the processor 180 provides a corresponding visual output on the display panel 141 according to the type of the touch event. Although, in FIG. 9, the touch-sensitive surface 131 and the display
panel 141 are used as two separate parts to implement input and output functions, in some embodiment, the touch-sensitive surface 131 and the display panel 141 may be integrated to implement the input and output functions.

[0079] The terminal 1200 may further include at least one sensor 150, such as an optical sensor, a motion sensor, and other sensors. Specifically, the optical sensor may include an ambient light sensor and a proximity sensor. The ambient light sensor may adjust luminance of the display panel 141 according to brightness of the ambient light. The proximity sensor may switch off the display panel 141 and/or backlight when the terminal 1200 is moved to the ear. As one type of motion sensor, a gravity acceleration sensor may detect magnitude of accelerations in various directions (generally on three axes), may detect magnitude and a direction of the gravity when static, and may be applied to an application that recognizes the attitude of a mobile phone (for example, switching between landscape orientation and portrait orientation, a related game, and magnetometer attitude calibration), a function related to vibration recognition (such as a pedometer and a knock), and the like. Other sensors, such as a gyroscope, a barometer, a hygrometer, a thermometer, and an infrared sensor, which may be configured in the terminal 1200, are not further described herein.

[0080] The audio circuit 160, a loudspeaker 161, and a microphone 162 may provide audio interfaces between the user and the terminal 1200. The audio circuit 160 may convert received audio data into an electric signal and transmit the electric signal to the loudspeaker 161. The loudspeaker 161 converts the electric signal into a sound signal for output. On the other hand, the microphone 162 converts a collected sound signal into an electric signal. The audio circuit 160 receives the electric signal and converts the electric signal into audio data, and outputs the audio data to the processor 180 for processing. Then, the processor 180 sends the audio data to, for example, another terminal device by using the RF circuit 110, or outputs the audio data to the memory 120 for further processing. The audio circuit 160 may further include an earplug jack, so as to provide communication between a peripheral earphone and the terminal 1200.

[0081] WiFi is a short distance wireless transmission technology. The terminal 1200 may help, by using the WiFi module 170, the user to receive and send e-mails, browse web pages, access stream media, and so on, which provides wireless broadband Internet access for the user. Although FIG. 9 shows the WiFi module 170, it may be understood that the WiFi module is not a necessary component of the terminal 1200, and when required, the WiFi module may be omitted as long as the scope of the essence of the present disclosure is not changed.
The processor 180 is the control center of the terminal 1200, and is connected to various parts of the mobile phone by using various interfaces and lines. By running or executing the software program and/or module stored in the memory 120, and invoking data stored in the memory 120, the processor 180 performs various functions and data processing of the terminal 1200, thereby performing overall monitoring on the mobile phone. Optionally, the processor 180 may include one or more processing cores. Preferably, the processor 180 may integrate an application processor and a modem processor. The application processor mainly processes an operating system, a user interface, an application programs, and the like, and the modem processor mainly processes wireless communication. It can be understood that the foregoing modem processor may not be integrated in the processor 180.

The terminal 1200 may further include the power supply 190 (such as a battery) for supplying power to the components. Preferably, the power supply may be logically connected to the processor 180 through a power management system, thereby implementing functions such as charging, discharging, and power consumption management by using the power supply management system. The power supply 190 may further include any component, such as one or more direct current or alternating current power supplies, a re-charging system, a power supply fault detection circuit, a power supply converter or an inverter, and a power supply state indicator.

Although not shown in the figure, the terminal 1200 may further include a camera, a Bluetooth module, and the like, which are not further described herein. Specifically, in this embodiment, the display unit of the terminal is a touch screen display, and the terminal further includes a memory and one or more programs. The one or more programs are stored in the memory, and are configured to be executed by one or more processors, where the one or more programs include instructions for executing the following operations:

determining, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response;

sending a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and displaying the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response; and

receiving the second-type action data sent by the server, and displaying the second-type action data.
Assuming that the foregoing is a first possible implementation manner, in a second possible implementation manner provided on the basis of the first possible implementation manner, the memory of the terminal further includes an instruction for executing the following operation:

listening, to determine whether the second-type action data is received.

In a third possible implementation manner provided on the basis of the second possible implementation manner, the memory in the terminal further includes an instruction for executing the following operation:

buffering the second-type action data if it is detected by means of listening that the second-type action data has arrived before display of the first-type action data is finished.

In a fourth possible implementation manner provided on the basis of the second possible implementation manner, the memory in the terminal further includes an instruction for executing the following operation:

continuing to display preset redundant action data associated with the first-type action data if it is detected by means of listening that the second-type action data has not arrived yet when display of the first-type action data is finished.

In conclusion, the terminal provided by this embodiment of the present invention determines, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response; sends a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and displays the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response; and receives the second-type action data sent by the server, and displays the second-type action data. Compared with the existing technology, when requesting the second-type action data from the server, the data processing method provided by this embodiment of the present invention may first display the first-type action data that can be displayed before the server makes a response, to avoid the feeling of stickiness during a game process, thereby improving communication efficiency between a user and a game application.

As another aspect, another embodiment of the present invention further provides a computer readable storage medium. The computer readable storage medium may be a computer readable storage medium included in the memory of the foregoing embodiment, or a separate computer readable storage medium that exists along and is not installed in a terminal. The computer
readable storage medium has one or more programs stored therein, where the one or more programs are executed by one or more processors to implement a data processing method, and the method includes:

determining, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response;

sending a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and displaying the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response; and

receiving the second-type action data sent by the server, and displaying the second-type action data.

[0090] Assuming that the foregoing is a first possible implementation manner, in a second possible implementation manner provided on the basis of the first possible implementation manner, when the first-type action data is displayed, the method further includes:

listening, to determine whether the second-type action data is received.

[0091] In a third possible implementation manner provided on the basis of the second possible implementation manner, the terminal buffers the second-type action data if it is detected by means of listening that the second-type action data has arrived before display of the first-type action data is finished.

[0092] In a fourth possible implementation manner provided on the basis of the second possible implementation manner, the terminal continues to display preset redundant action data associated with the first-type action data if it is detected by means of listening that the second-type action data has not arrived yet when display of the first-type action data is finished.

[0093] FIG. 10 is a schematic structural diagram of a server 50 according to an embodiment of the present invention. The server 50 may include an input device 510, an output device 520, a processor 530, and a memory 540.

[0094] The memory 540 may include a read-only memory (ROM) and a random access memory (RAM), and provides instructions and data for the processor 530. A part of the memory 540 may further include a non-volatile random access memory (NVRAM).
The memory 540 stores the following elements, and can execute modules or data structures, or a subset thereof, or an extended set thereof:

- an operation instruction: including various operation instructions for implementing various operations; and

- an operating system: including various system programs for implementing basic services and processing hardware-based tasks.

In this embodiment of the present invention, the processor 530 invokes an operation instruction stored in the memory 540 (where the operation instruction may be stored in the operating system), to execute the following operations:

- receiving, by the input device 51, a data requesting message sent by a terminal when the terminal displays first-type action data, the data requesting message being used for requesting second-type action data corresponding to a specified skill specified by a user, the first-type action data being data that can be displayed before the server makes a response, and the second-type action data being data that cannot be displayed until the server makes a response; determining, according to the data requesting message, the second-type action data corresponding to the specified skill; and

- sending, by the output device 520, the second-type action data to the terminal.

In this embodiment of the present invention, when requesting the second-type action data from the server, the terminal may first display the first-type action data that can be displayed before the server makes a response, to avoid the feeling of stickiness during a game process, thereby improving communication efficiency between a user and a game application.

The processor 530 controls operation of the server 50, and the processor 530 may also be referred to as a CPU. The memory 540 may include a ROM and a RAM, and provides instructions and data for the processor 530. A part of the memory 540 may further include an NVRAM. In a specific application, the components of the server 50 may be coupled with each other by using a bus system 550, where in addition to a data bus, the bus system 550 may further include a power supply bus, a control bus, a status signal bus, and the like. For the clarity of description, all buses are marked as the bus system 550 in the figure.

The method disclosed in the foregoing embodiment of the present invention may be applied to the processor 530, or implemented by the processor 530. The processor 530 may be an integrated circuit chip and has a signal processing capability. In an implementation process, each of the foregoing method may be implemented by an integrated logic circuit of hardware in the
processor 530 or by a software-form instruction. The processor 530 may be a general-purpose processor, a digital signal processor (DSP), an application specific integrated circuit (ASIC), a field programmable gate array (FPGA) or another programmable logic device, a discrete gate or transistor logic device, or a discrete hardware component, and can implement or execute each method, step, and logic block diagram disclosed in the embodiments of the present invention. The general-purpose processor may be a microprocessor; alternatively, the processor may also be any regular processor, or the like. Steps of the method disclosed with reference to the embodiment of the present invention may be directly implemented by a hardware decoding processor, or implemented by a combination of hardware in the decoding processor and a software module. The software module may be located in a mature storage medium in the field, such as a RAM, a flash memory, a ROM, a programmable read only memory or an electrically erasable programmable memory, or a register. The storage medium is located in the memory 540, and the processor 530 reads information in the memory 540, to implement the foregoing method steps together with hardware.

[0100] Optionally, the memory 540 is configured to buffer the data requesting message when the processor 530 determines that the data requesting message arrives at the server ahead of time; and

the processor 530 is configured to record an expiration time of the data requesting message, and reads the data requesting message when the expiration time arrives.

[0101] Referring to FIG. 11, an embodiment of a data processing system provided by the embodiments of the present invention includes a terminal 40 and a server 50.

[0102] The terminal 40 is configured to determine, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed without waiting for a response of a server; send a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill; and display the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response.

[0103] The server 50 is configured to receive the data requesting message sent by the terminal, determine the second-type action data corresponding to the specified skill according to the data requesting message, and send the second-type action data to the terminal.

[0104] The terminal 40 is configured to receive the second-type action data sent by the server, and display the second-type action data.
A person of ordinary skill in the art can understand that all or some steps of the methods according to the foregoing embodiments may be implemented by a program instructing relevant hardware. The program may be stored in a computer readable storage medium, and the storage medium may include a ROM, a RAM, a magnetic disk, an optical disc, or the like.

The data processing method, terminal, server, and system provided by the embodiments of the present invention are described in detail above. Specific examples are used herein to illustrate principles and implementation manners of the present disclosure. The above description of the embodiments is merely for helping understand the method and core idea of the present disclosure. Meanwhile, a person of ordinary skill in the art may make alterations to the specific implementation manner and application range according to the idea of the present disclosure. In conclusion, the content of the specification shall not be construed as a limitation to the present disclosure.
What is claimed is:

1. A data processing method, comprising:

   at a terminal having one or more processors and memory storing programs executed by the one or more processors,

   determining, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response;

   sending a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and displaying the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response; and

   receiving the second-type action data sent by the server, and displaying the second-type action data.

2. The method according to claim 1, wherein when the first-type action data is displayed, the method further comprises:

   listening, to determine whether the second-type action data is received.

3. The method according to claim 2, wherein,

   buffering the second-type action data if it is detected by means of listening that the second-type action data has arrived before display of the first-type action data is finished.

4. The method according to claim 2, wherein,

   continuing to display preset redundant action data associated with the first-type action data if it is detected by means of listening that the second-type action data has not arrived yet when display of the first-type action data is finished.

5. A data processing method, comprising:

   at a server having one or more processors and memory storing programs executed by the one or more processors,

   receiving a data requesting message sent by a terminal when the terminal displays first-type action data, the data requesting message being used for requesting second-type action data
corresponding to a specified skill specified by a user, the first-type action data being data that can be displayed before a server makes a response, and the second-type action data being data that cannot be displayed until the server makes a response;

determining, according to the data requesting message, the second-type action data corresponding to the specified skill; and

sending the second-type action data to the terminal.

6. The method according to claim 5, wherein after the receiving a data requesting message sent by the terminal, the method further comprises:

buffering the data requesting message and recording an expiration time of the data requesting message when determining that the data requesting message arrives at the server ahead of time; and

reading the data requesting message when the expiration time arrives.

7. A terminal, comprising processors and various program modules storing processor-executable instructions, the program modules comprising:

a receiving unit, configured to receive a casting instruction input by a user for a specified skill;

a determining unit, configured to determine first-type action data corresponding to the specified skill received by the receiving unit, the first-type action data being data that can be displayed before a server makes a response;

a sending unit, configured to send a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill received by the receiving unit; and

a display unit, configured to display the first-type action data determined by the determining unit, the second-type action data being data that cannot be displayed until the server makes a response;

the receiving unit being further configured to receive the second-type action data sent by the server; and

the display unit being further configured to display the second-type action data received by the receiving unit.

8. The terminal according to claim 7, further comprising:

a listening unit, configured to listen, when the display unit displays the first-type action data, to
determine whether the second-type action data is received.

9. The terminal according to claim 8, further comprising:

a buffering unit, configured to buffer the second-type action data if the listening unit detects by means of listening that the second-type action data has arrived before display of the first-type action data is finished.

10. The terminal according to claim 8, wherein

the display unit is further configured to continue to display preset redundant action data associated with the first-type action data if the listening unit detects by means of listening that the second-type action data has not arrived yet when display of the first-type action data is finished.

11. A server, comprising:

a receiving unit, configured to receive a data requesting message sent by a terminal when the terminal displays first-type action data, the data requesting message being used for requesting second-type action data corresponding to a specified skill specified by a user, the first-type action data being data that can be displayed before the server makes a response, and the second-type action data being data that cannot be displayed until the server makes a response;

a determining unit, configured to determine, according to the data requesting message received by the receiving unit, the second-type action data corresponding to the specified skill; and

a sending unit, configured to send, to the terminal, the second-type action data determined by the determining unit.

12. The server according to claim 11, further comprising:

a buffering unit, configured to buffer the data requesting message when the determining unit determines that the data requesting message arrives at the server ahead of time;

a recording unit, configured to record an expiration time of the data requesting message buffered by the buffering unit; and

a reading unit, configured to read the data requesting message buffered by the buffering unit when the expiration time recorded by the recording unit arrives.
Determine, after receiving a casting instruction input by a user for a specified skill, first-type action data corresponding to the specified skill, the first-type action data being data that can be displayed before a server makes a response

Send a data requesting message to the server, the data requesting message being used for requesting second-type action data corresponding to the specified skill, and display the first-type action data, the second-type action data being data that cannot be displayed until the server makes a response

Receive the second-type action data sent by the server, and display the second-type action data

FIG. 1

Receive a data requesting message sent by a user terminal when the user terminal displays first-type action data, the data requesting message being used for requesting second-type action data corresponding to a specified skill specified by a user, the first-type action data being data that can be displayed before a server makes a response, and the second-type action data being data that cannot be displayed until the server makes a response

Determine, according to the data requesting message, the second-type action data corresponding to the specified skill

Send the second-type action data to the user terminal

FIG. 2
S300. Receive a casting instruction input by a user for a specified skill

S305. Determine first-type action data corresponding to the specified skill

S310. Send a data request message to the server

S315. Display the first-type action data

S320. Buffer the data requesting message when the data requesting message arrives ahead of time

S325. Determine the second-type action data according to the data requesting message

S330. Second-type action data

S335. Buffer the second-type action data when the second-type action data arrives ahead of time

S340. Continue to display preset redundant action data associated with the first-type action data when the second-type action data arrives later than expected

S345. Display the second-type action data

FIG. 3
FIG. 4

FIG. 5
A. **CLASSIFICATION OF SUBJECT MATTER**

G06F 19/00(2011.01)i

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)

G06F

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

Electronic database consulted during the international search (name of database and, where practicable, search terms used)

WPI, EPDOC, CNKI, CNPAT: action, part, type, cast, server, display, delay, data, online, game, instruction, skill, response, buffer

C. **DOCUMENTS CONSIDERED TO BE RELEVANT**

<table>
<thead>
<tr>
<th>Category*</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PX</td>
<td>CN 103886198 A (TENCENT TECHNOLOGY SHENZHEN COMPANY LIMITED) 25 June 2014 (2014-06-25) claims 1-12</td>
<td>1-12</td>
</tr>
<tr>
<td>A</td>
<td>CN 103501287 A (NETEASE HANGZHOU NETWORK CO., LTD.) 08 January 2014 (2014-01-08) the whole document</td>
<td>1-12</td>
</tr>
</tbody>
</table>

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
  - "D": 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
  - "Q": document member of the same patent family

Date of the actual completion of the international search: 30 April 2015

Date of mailing of the international search report: 03 June 2015

Name and mailing address of the ISA/CN:

STATE INTELLECTUAL PROPERTY OFFICE OF THE P.R.CHINA
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088, China

Authorized officer: HU, Ping

Facsimile No. (86-10)62019451

Telephone No. (86-10)82245429
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date (day/month/year)</th>
<th>Patent family member(s)</th>
<th>Publication date (day/month/year)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CN 103886198 A</td>
<td>25 June 2014</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>US 2013205220 A1</td>
<td>08 August 2013</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>US 7346439 B2</td>
<td>18 March 2008</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>CN 103501287 A</td>
<td>08 January 2014</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>